



The
University
Of
Sheffield.

A single molecule approach to studying magnesium chelatase

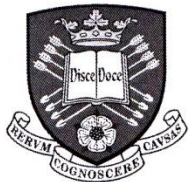
By:

Dawn Hesson

A thesis submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

The University of Sheffield
Faculty of Science
Department of Chemistry

September 2015



The
University
Of
Sheffield.

Access
To
Thesis.

This thesis is protected by the Copyright, Designs and Patents Act 1988. No reproduction is permitted without consent of the author. It is also protected by the Creative Commons Licence allowing Attributions-Non-commercial-No derivatives.

- A bound copy of every thesis which is accepted as worthy for a higher degree, must be deposited in the University of Sheffield Library, where it will be made available for borrowing or consultation in accordance with University Regulations.
- All students registering from 2008-09 onwards are also required to submit an electronic copy of their final, approved thesis. Students who registered prior to 2008-09 may also submit electronically, but this is not required.

Author: DAWN HESSON Dept: CHEMISTRY

Thesis Title: A single molecule approach to studying magnesium chelataze. Registration No: 100132654

For completion by all students:

Submit in print form only (for deposit in the University Library):

Submit in print form and also upload to the White Rose eTheses Online server: In full ☐

Edited eThesis ☐

Please indicate if there are any embargo restrictions on this thesis. Please note that if no boxes are ticked, you will have consented to your thesis being made available without any restrictions.

Embargo details: (complete only if requesting an embargo to either your print and/or eThesis)

Embargo required?

Length of embargo
(in years)

Print Thesis Yes ☒ No ☐

eThesis Yes ☒ No ☐

5
5

Supervisor: I, the supervisor, agree to the named thesis being made available under the conditions specified above.

Name: R REID Dept: Chemistry

Signed: [Signature] Date: 22/9/15.

Student: I, the author, agree to the named thesis being made available under the conditions specified above.

I give permission to the University of Sheffield to reproduce the print thesis in whole or in part in order to supply single copies for the purpose of research or private study for a non-commercial purpose.

I confirm that this thesis is my own work, and where materials owned by a third party have been used copyright clearance has been obtained. I am aware of the University's *Guidance on the Use of Unfair Means* (www.sheffield.ac.uk/lets/design/unfair)

I confirm that all copies of the thesis submitted to the University (including electronic copies on CD/DVD) are identical in content.

Name: DAWN HESSON Dept: CHEMISTRY

Signed: [Signature] Date: 22/9/15

For completion by students also submitting an electronic thesis (eThesis):

I, the author, agree that the University of Sheffield's eThesis repository (currently WREO) will make my eThesis available over the internet via an entirely non-exclusive agreement and that, without changing content, WREO may convert my thesis to any medium or format for the purpose of future preservation and accessibility.

I, the author, agree that the metadata relating to the eThesis will normally appear on both the University's eThesis server and the British Library's ETHOS service, even if the thesis is subject to an embargo. I agree that a copy of the eThesis may be supplied to the British Library.

I confirm that the upload is identical to the final, examined and awarded version of the thesis as submitted in print to the University for deposit in the Library (unless edited as indicated above).

Name: DAWN HESSON Dept: CHEMISTRY

Signed: [Signature] Date: 25/9/15

THIS SHEET MUST BE BOUND IN THE FRONT OF THE PRINTED THESIS BEFORE IT IS SUBMITTED

Abstract

The enzyme magnesium chelatase catalyses the insertion of magnesium into protoporphyrin IX to form magnesium protoporphyrin IX in the first committed step of chlorophyll biosynthesis. It comprises three different subunits known as ChII, ChID and ChIH. ChII and ChID interact in an ATP-dependent manner, forming a complex to which the porphyrin-carrying subunit ChIH binds. Magnesium insertion requires a large amount of energy provided by the concomitant ATP hydrolysis of the ChII subunit.

A wide range of experimental methods have been employed to determine the stoichiometry and size of the active ID complex, but no general consensus has as yet been reached. Single molecule subunit counting is another, complementary method with which to address the problem. By labelling the subunits of the ID complex with fluorescent molecules, it has been possible to monitor the numbers of subunits within individual complexes directly from the fluorescence intensity of the labels. This technique reveals that the form of the ID complex is dynamic and the number of ChID subunits involved in the ID complex varies between 1 and 5.

Previous work has investigated the ATPase activity and nucleotide binding kinetics of the ChII subunit in ensemble measurements. This accessed the behaviour of the free enzyme species and determined that ChII binds nucleotide according to the mechanism of conformational selection. Single molecule nucleotide binding experiments have probed the interaction of fluorescent nucleotide with the ChII subunit directly. These provide a value for the off-rate binding constant of $6.0 \pm 1.9 \text{ s}^{-1}$ which suggests either that it is inappropriate to use the rapid equilibrium approximation when interpreting transient kinetic data for this subunit, or that the enzyme undergoes a second isomerization upon binding nucleotide.

Acknowledgements

Funding for this work was provided by Research Councils UK through the E-Futures programme.

I would like to thank Dr Ashley Cadby and Dr Jim Reid for giving me the opportunity to complete this work and for their assistance in my development as a scientist.

I would also like to thank Dr Nate Adams for his assistance in developing my knowledge of this field and practical abilities, Sijing Xia for making the surfaces used in this work.

For their support I would like to thank my colleagues from the Department of Chemistry particularly Luke Taurino, Zeyed Abdulkarim and P.J. Taylor.

For their support I would like to thank my colleagues from the Department of Physics and Astronomy particularly Dr Bob Turner for his image analysis code which was built upon in this work.

Finally I would like to thank my family for their love and support throughout.

Contents

List of Figures	v
List of Schemes.....	ix
List of Tables	xi
Abbreviations.....	xiii
1. Introduction	1
1.1. Magnesium chelatase	1
1.1.1. The role of magnesium chelatase within the context of photosynthesis.....	1
1.1.1.1. Photosynthesis.....	1
1.1.1.2. Chlorophyll biosynthesis	2
1.1.2. Structure and function of the subunits of magnesium chelatase.....	4
1.1.2.1. The I subunit	4
1.1.2.1.1. Primary and tertiary structure	4
1.1.2.1.2. ATPase activity	5
1.1.2.1.3. Oligomeric structure	6
1.1.2.2. The D subunit	8
1.1.2.2.1. Primary structure	8
1.1.2.2.2. Bchl is required for correct folding of BchD.....	8
1.1.2.3. The ID complex.....	8
1.1.2.4. The H Subunit.....	9
1.1.2.4.1. Primary and tertiary structure	9
1.1.2.4.2. Porphyrin binding.....	11
1.1.3. The magnesium chelatase reaction	11
1.1.3.1. Optimum subunit proportions.....	11
1.1.3.2. Substrate requirements and kinetic analyses.....	11
1.1.4. Summary	13
1.2. Single molecule microscopy.....	14
1.2.1. Fluorescence microscopy.....	14
1.2.1.1. The phenomenon of fluorescence	14
1.2.1.2. The fluorescence microscope	18
1.2.1.3. The diffraction limit and resolution	19
1.2.2. Total internal reflection	20
1.2.2.1. Prism-type total internal reflection	22
1.2.2.2. Objective-type total internal reflection	22

1.2.3. Image analysis techniques	23
1.2.3.1. Fluorophore localization	23
1.2.3.2. Step analysis.....	25
1.2.4. Determining protein quaternary structure.....	26
1.2.5. Nucleotide binding and dynamics studies	28
1.3. Aims.....	28
2. Materials and methods for biochemical analyses	31
2.1. Materials	31
2.1.1. Chemicals and reagents	31
2.1.2. <i>E. coli</i> strains and plasmids	31
2.2. Preparation of growth media, reagents and buffers	31
2.3. Transformation of chemically competent <i>E. coli</i>	33
2.4. Propagation of plasmid DNA.....	34
2.5. Expression of magnesium chelatase subunits	34
2.5.1. ChII, His ₆ -ChII and ChIH	34
2.5.2. ChID.....	34
2.6. SDS-polyacrylamide gel electrophoresis.....	34
2.7. Purification of magnesium chelatase subunits	35
2.7.1. His ₆ -tagged proteins.....	35
2.7.2. Non-tagged proteins	36
2.8. Protein, fluorescent dye and substrate concentration determination.....	37
2.8.1. UV-Vis absorbance	37
2.8.2. Bradford Assay	39
2.9. Magnesium chelatase assay.....	39
2.9.1. Assay conditions.....	39
2.9.2. Data analysis	39
2.10. ATPase assay	40
2.11. Fluorescent nucleotide titrations.....	40
2.11.1. TNP-ATP titration assay.....	40
2.11.2. Cy3-ATP competition titration	41
2.12. Pull downs.....	41
2.13. Fluorescent labelling of protein.....	42
2.13.1. Fluorescent dye preparation.....	42
2.13.2. Protected single labelling of ChII	42

2.13.3. Labelling ChID.....	42
2.13.4. Desalting to remove unreacted dye.....	43
2.13.5. Checking conjugation and determining labelling efficiency	43
2.14. Purification of Cy3-EDA-ATP	43
2.15. Single molecule studies.....	43
2.15.1. Stoichiometry measurements.....	43
2.15.2. Nucleotide binding assays.....	44
3. Methods and experimental setup for single molecule microscopy.....	45
3.1. Equipment and materials	45
3.2. Preparation of microscope slides.....	46
3.2.1. Cleaning.....	47
3.2.2. Surface modification	47
3.2.3. Sample preparation.....	47
3.3. Optical arrangement for single molecule microscopy	48
3.3.1. Stoichiometry measurements.....	48
3.3.2. Nucleotide binding measurements.....	52
3.4. Data acquisition	53
4. Analysis of microscope images	55
4.1. Localization of fluorophores	55
4.2. Stepwise photobleaching analysis for stoichiometry	58
4.2.1. Refinement of fluorophore list and intensity extraction	58
4.2.2. Determination of complex stoichiometry.....	61
4.3. 'On' and 'off' time analysis for fluorescent nucleotide binding.....	64
4.3.1. Refinement of fluorophore list and intensity extraction	64
4.3.2. Extraction of event durations and amplitudes	65
4.3.3. Maximum likelihood analysis to extract time components.....	70
4.3.4. Distinguishing between events of interest and background events.....	71
5. The stoichiometry of ChID in the ID complex	73
5.1. Labelling subunits through their cysteine residues	74
5.1.1. Determining labelling success and efficiency	76
5.1.2. Activity of labelled subunits.....	77
5.2. Determining optimum protein concentration on a coverslip.....	81
5.3. Distinguishing between labelled subunits and background fluorescence.....	82
5.4. Investigating the stoichiometry of ChID in the ID complex	83

6. Interaction of fluorescent nucleotide with the ID complex.....	95
6.1. TNP-ATP	95
6.1.1. TNP-ATP binds to ChII and can sustain an ID interaction	97
6.1.2. TNP-ATP is an inhibitor of the chelatase reaction	99
6.1.3. Single molecule visualization of TNP-ATP binding to chelatase subunits.....	102
6.2. Cy3-EDA-ATP	104
6.2.1. Separation of Cy3-EDA-ATP isomers.....	105
6.2.2. Cy3-EDA-ATP binds to ChII	109
6.2.3. Single molecule visualization of Cy3-EDA-ATP interaction with His ₆ -ChII	112
7. Summary	121
7.1. Counting ChID subunits in the ID complex.....	121
7.2. Single molecule nucleotide binding to ChII.....	122
7.3. Conclusions and future work	123
References	125
Appendix A: MATLAB code for localization of fluorophores	135
Appendix B: MATLAB code for analysis of stepwise photobleaching intensity traces	149
Appendix C: MATLAB code for analysis of fluorescent nucleotide binding.....	173

List of Figures

Figure 1.1. Electron micrograph of a chloroplast.	2
Figure 1.2. An outline of illustrative tetrapyrrole biosynthetic pathways.....	2
Figure 1.3. The branch point between the haem and chlorophyll biosynthetic pathways.	3
Figure 1.4. Structures of a typical AAA ⁺ domain and Bchl.	5
Figure 1.5. Oligomeric configuration of AAA ⁺ domains.	7
Figure 1.6. 3D models of ChlH and BchH with and without protoporphyrin IX substrate bound.	10
Figure 1.7. Electronic energy level diagrams illustrating photophysical processes.....	16
Figure 1.8. Fluorescence excitation and emission spectra.	17
Figure 1.9. Principle of operation of an inverted fluorescence microscope using epi-illumination.	19
Figure 1.10. Distinguishing two point source emitters.....	20
Figure 1.11. Schematics of refraction and total internal reflection.....	21
Figure 1.12. Schematic showing one possible optical arrangement for prism-type total internal reflection	22
Figure 1.13. Schematic showing the optical arrangement for objective-type total internal reflection.....	23
Figure 3.1. Schematic representation of surface modification.	46
Figure 3.2. Schematic showing the construction of a microscope sample.....	48
Figure 3.3. Schematic of the optical bench setup used in experiments investigating the stoichiometry of the ChlI-ChlD complex.	49
Figure 3.4. Schematic showing the illumination configuration within the microscope in (A) epi-fluorescence mode and (B) TIR mode.	50
Figure 3.5. Schematic of the optical bench layout for experiments investigating nucleotide binding to ChlI.	52
Figure 4.1. Summary of steps required to localize fluorophores within an image series.	55
Figure 4.2 Background removal.	56
Figure 4.3. Fluorophore localization.	57
Figure 4.5. Summary of steps required to refine the list of localized fluorophores and extract their intensities over an image series. The custom MATLAB functions used in each step are listed below the description.	58
Figure 4.4. Fluorophores are tracked through every frame of an image series.	58
Figure 4.6. Overlapping fluorophores are discarded.	59
Figure 4.7. Sample photobleaching steps in the intensity-time trace of a fluorescently labelled protein complex.	60
Figure 4.8. Determination of single fluorophore intensity.	60
Figure 4.9. Summary of steps required to infer stoichiometry from stepwise photobleaching intensity traces. The custom MATLAB functions used in each step are listed below the description.	61
Figure 4.10. The Chung Kennedy (CK) filter smooths noise in the intensity trace while preserving the edges of abrupt transitions in intensity level.	62
Figure 4.11. Sample pairwise distance distribution function highlights the discrete intensity levels in the stepwise photobleaching trace of a fluorescently labelled protein complex. ...	63

Figure 4.13. Summary of steps required to refine the list of localized fluorophores and extract their intensities over an image series. The custom MATLAB functions used in each step are listed below the description.	64
Figure 4.12. Peak identification in a pairwise distance distribution function (PDDF).	64
Figure 4.14. Intensity as a function of time for fluorescently labelled nucleotide binding to and dissociating from a ChII molecule.	65
Figure 4.15 Summary of steps required to extract binding event durations and their amplitudes over an image series. The custom MATLAB functions used in each step are listed below the description.	65
Figure 4.16. Chung Kennedy (CK) filtering increases the signal to noise ratio and reduces the number of false events detected.	66
Figure 4.17. Response of Chung Kennedy filter to short duration events.	68
Figure 4.18. Half amplitude threshold crossing analysis enables identification of binding events.	69
Figure 4.20. Summary of steps required for maximum likelihood analysis. The custom MATLAB functions used in each step are listed below the description.	70
Figure 4.19. Imposing a resolution limit on event durations removes events which are shorter than the Chung Kennedy (CK) filter dead time.	70
Figure 4.21. Histogram of binding event durations for a single diffraction-limited area.	71
Figure 5.1. Amino acid sequence alignment of ChII and ChID.	75
Figure 5.2. Maleimide modification of a thiol group to form a thioether bond.	76
Figure 5.3. SDS-PAGE gels demonstrating ChII and ChID subunits labelled with fluorescent dyes.	76
Figure 5.4. Example calibration and progress curves from a magnesium chelatase assay. ..	78
Figure 5.5. Chemical structures of thiol-reactive modifiers N-ethyl maleimide and Cy5 maleimide.	80
Figure 5.6. Fluorescently-labelled ChII binding to ChID.	80
Figure 5.7. Effect of varying protein concentration on the density of fluorescent molecules visible on a coverslip.	81
Figure 5.8. Effect of varying wash buffer volume on the density of fluorescent molecules visible on a coverslip.	82
Figure 5.9. The fluorescence emission from immobilized fluorophores is visible above background fluorescence intrinsic to the sample coverslip.	83
Figure 5.10. Example photobleaching intensity traces.	85
Figure 5.11. Histograms of single molecule photobleaching steps for ChID multiply labelled with Cy5 in complex with ChII.	87
Figure 5.12. Histograms of single molecule photobleaching steps for ChID singly labelled with Cy3 in complex with ChII.	89
Figure 5.13. Histogram of single molecule photobleaching steps for ChID singly labelled with Cy3 in complex with ChII.	91
Figure 5.14. Simulated data of complexes comprising 3 subunits based on the fluorophore lifetimes and SNRs of the data shown in A) Figure 5.11 with a 15 % chance that two complexes will overlap, and B) Figure 5.12 with a 17 % chance that two complexes will overlap.	92
Figure 6.1. Chemical structure of TNP-ATP.	96

Figure 6.2. The fluorescence emission of TNP-ATP increases upon binding to ChII.	96
Figure 6.3. Titration assay to determine the dissociation constant (K_d) for TNP-ATP binding to NHis ₆ -ChII.	98
Figure 6.4. Co-purification of ChII and His ₆ -ChID in the presence of TNP-ATP.	99
Figure 6.5. MgD _{IX} standard curves in the presence of TNP-ATP.	100
Figure 6.6. Plots of steady state rates against substrate concentration for chelatase assays in the presence of TNP-ATP.	100
Figure 6.7. Plots of A) V_{max} and B) V_{max}/K_m against concentration of TNP-ATP for chelatase assays.	101
Figure 6.8. ID complexes immobilized on a coverslip in the presence of TNP-ATP.	103
Figure 6.9. Chemical structure of Cy3-EDA-ATP	104
Figure 6.10. Separation of Cy3-EDA-ATP isomers via HPLC.	105
Figure 6.11. Purity of Cy3-EDA-ATP isomer I after evaporation with addition of methanol <i>in vacuo</i>	106
Figure 6.12. Purity of Cy3-EDA-ATP isomers after drying.	107
Figure 6.13. HPLC absorbance trace showing isomers of Cy3-EDA-ATP and Cy3-EDA-ADP.	108
Figure 6.14. Check for presence of 3'-O-Cy3-EDA-ADP in purified 3'-O-Cy3-EDA-ATP.	109
Figure 6.15. Fluorescence emission spectra for competition titration between TNP-ATP and Cy3-ATP in the presence of ChII.	110
Figure 6.16. Total fluorescence emission for competition titration between TNP-ATP and Cy3-ATP in the presence of ChII.	111
Figure 6.17. Simulations for varying $K_{d,Cy3-ATP}$. All models were generated using Dynafit[107] with the value for $K_{d,TNP-ATP}$ fixed at 3.5 μ M.	112
Figure 6.18. Background fluorescence from Cy3-ATP in solution.	113
Figure 6.19. Average lifetime distributions for N-terminally tagged ChII in normal chelatase buffer.	115
Figure 6.20. Average lifetime distributions for C-terminally tagged ChII in normal chelatase buffer.	116
Figure 6.21. Average lifetime distributions for N-terminally tagged ChII in high salt chelatase buffer.	116
Figure 6.22. Average lifetime distributions for C-terminally tagged ChII in high salt chelatase buffer.	117
Figure 7.1. Possible ID complex stoichiometries present in single molecule photobleaching experiments.	122

List of Schemes

Scheme 1.1. Possible reaction pathways for ATP hydrolysis by magnesium chelatase.	6
Scheme 1.2. Reaction scheme for magnesium chelatase in the presence of saturating Mg^{2+}	13
Scheme 1.3. Possible reaction pathways for metal ion chelation.	13
Scheme 2.1. Competitive binding mechanism.....	41
Scheme 6.1. Isomerization of enzyme before substrate binding.	118
Scheme 6.2. Isomerization of enzyme before and after binding substrate.	118

List of Tables

Table 2.1. Molecular masses and extinction coefficients of magnesium chelatase subunits	37
Table 2.2. Molecular masses and extinction coefficients of fluorescent dyes and substrates.	38
Table 3.1. Details of equipment used in the optical bench setup for single molecule microscopy experiments	45
Table 5.1. Mean labelling efficiencies for ChII and ChID subunits labelled with fluorescent dyes.	77
Table 5.2. Mean steady state rates of magnesium chelatase assays performed at saturating substrate concentrations.	79
Table 5.3. Fitting parameters for the binomial distributions used to describe the histograms of single molecule photobleaching steps shown in Figure 5.10. n: number of subunits in a complex, θ : probability of detecting a fluorescent step for a particular fluorescent label.	87
Table 5.4. Fitting parameters for the binomial distributions used to describe the histograms of single molecule photobleaching steps shown in Figure 5.11. n: number of subunits in a complex, θ : probability of detecting a fluorescent step for a particular fluorescent label, A: proportion of complexes which do not overlap.	90
Table 5.5. Fitting parameters for the binomial distributions used to describe the histograms of single molecule photobleaching steps shown in Figure 5.12. n: number of subunits in a complex, θ : probability of detecting a fluorescent step for a particular fluorescent label, A: proportion of complexes which do not overlap.	91

Abbreviations

2-PDS	2,2'-dipyridyl disulphide
AB-NTA	N _α ,N _α -bis(carboxymethyl)-L-lysine hydrate
ADP	Adenosine 5'-diphosphate
AEBSF	4-(2-aminoethyl)benzenesulfonyl fluoride hydrochloride
ALA	5-aminolevulinic acid
APTES	(3-aminopropyl)triethoxysilane
ATP	Adenosine 5'-triphosphate
Bchl	Bacteriochlorophyll
BSA	Bovine serum albumin
CD	Circular dichroism
Chl	Chlorophyll
Cy3-EDA-ATP	2'/3'-O-(2-aminoethyl-carbamoyl)-adenosine-5'-triphosphate, labelled with Cy3, triethylammonium salt
D _{IX}	Deuteroporphyrin IX
DMF	Dimethylformamide
DMSO	Dimethyl sulfoxide
DTT	Dithiothreitol
EDTA	ethylenediaminetetraacetic acid
EMCCD	Electron multiplying charge coupled device
FP	Fluorescent protein
FPLC	Fast protein liquid chromatography
GFP	Green fluorescent protein
IMAC	Immobilized metal affinity chromatography
IPTG	Isopropyl-β-D-thiogalactopyranoside
LDH	lactate dehydrogenase
MgD _{IX}	Magnesium deuteroporphyrin IX
MOPS	3-(N-morpholino)propanesulfonic acid
NA	Numerical aperture
NADH	Nicotinamide adenine dinucleotide
NEM	N-ethyl maleimide
NTA	Nitrilotriacetic acid
PEP	Phospho(enol)pyruvate
PFS	Perfect Focus System
PIPES	Piperazine-1,4-bis(2-ethanesulfonic acid)
P _{IX}	Protoporphyrin IX
PK	Pyruvate kinase
PSF	Point spread function
SAM	Self-assembled monolayer
sCMOS	Scientific complementary metal-oxide semiconductor
SDS-PAGE	Sodium dodecyl sulphate polyacrylamide gel electrophoresis
SNR	Signal-to-noise ratio
TCE	2,2,2-trichloroethanol
TEAB	Triethylammonium bicarbonate
TIF	Tagged image file
TIR	Total internal reflection
TIRFM	Total internal reflection fluorescence microscopy
TNP-ATP	2'-(or-3')-O-(trinitrophenyl)adenosine 5'-triphosphate
Tris	Tris(hydroxymethyl)aminomethane

1. Introduction

1.1. Magnesium chelatase

Magnesium chelatase catalyses a step in the biosynthesis of the photosynthetic pigment chlorophyll, a multi-step process which shares precursor molecules with other tetrapyrrole-derived compounds such as the haems, cobalamines and phycobilins. The chelatase reaction is located at the branch point between the chlorophyll and haem biosynthetic pathways (Figure 1.3). Magnesium chelatase is a complex enzyme consisting of three different types of subunits, two of which belong to the AAA⁺ (ATPases associated with various cellular activities) superfamily of proteins.

1.1.1. The role of magnesium chelatase within the context of photosynthesis

1.1.1.1. Photosynthesis

Photosynthesis is the process by which organisms harvest energy from sunlight and convert it to chemical forms which can be stored and used to drive cellular activities. Essentially all life on Earth depends on the energy captured and stored by photosynthetic organisms. They provide food for all heterotrophs (organisms which derive energy from the breakdown of organic material), and most of our energy resources[1].

The most obvious and well-known form of photosynthesis is chlorophyll-based (referred to simply as photosynthesis from this point). It relies on the group of macrocyclic tetrapyrrole pigments known as chlorophylls to absorb light efficiently and perform the first step in converting the absorbed solar energy to chemical form. Green chlorophylls are evident as the pigments giving the green colour to much of the plant life on Earth. There are nearly 100 different chlorophyll structures known currently, which can be divided into two main categories: chlorophylls (Chl) and bacteriochlorophylls (Bchl). Chlorophylls are produced by oxygenic organisms (including eukaryotic algae and plants, and the single oxygenic group of prokaryotes, the cyanobacteria) which evolve oxygen as a product of photosynthesis, while bacteriochlorophylls are synthesized by anoxygenic organisms (including purple bacteria, green sulphur bacteria, filamentous anoxygenic phototrophs, heliobacteria, and chloroacidobacteria)[1,2].

In eukaryotic organisms, photosynthesis and chlorophyll production occur in specialized organelles known as chloroplasts (a type of plastid). These are located in the cells of leaves in higher plants. Figure 1.1 shows an electron micrograph of a chloroplast. Within the chloroplast, thylakoid membranes form an extensive network, assembling themselves into stacks known as grana. The aqueous medium filling the chloroplast and surrounding the membranes is the stroma. Many of the proteins involved in photosynthesis are membrane proteins associated with the thylakoid membrane. Prokaryotic photosynthetic cells do not possess chloroplasts, but some of the cell's cytoplasmic membrane is adapted to form thylakoid membranes which provide the site where the early steps of photosynthesis take place[1,3].

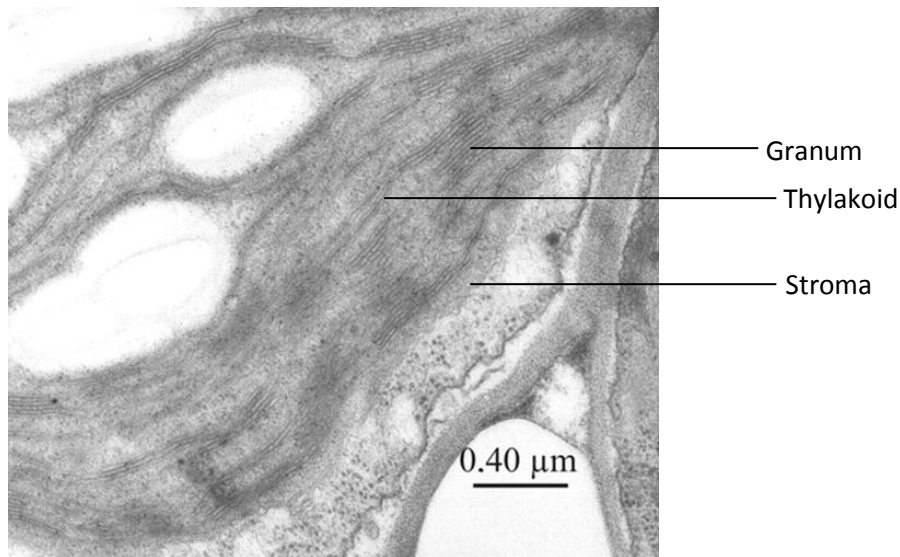


Figure 1.1. Electron micrograph of a chloroplast.

This image was originally published in the Journal of Biological Chemistry. Jana Stöckel and Ralf Oelmüller. A Novel Protein for Photosystem I Biogenesis. J. Biol. Chem. 2004; 279:10243-10251. © the American Society for Biochemistry and Molecular Biology[134]

1.1.1.2. Chlorophyll biosynthesis

Chlorophylls (Chl and Bchl) are all derived from the precursor tetrapyrrole molecule, protoporphyrin IX (P_{IX}). Tetrapyrrole-derived compounds are widely used in nature for processes ranging from photosynthesis (chlorophylls) and respiration (haems) to methanogenesis (coenzyme F_{430}). The chlorophyll biosynthetic pathway is just one of many branches in the network of pathways leading to tetrapyrrole cofactors and prosthetic groups (Figure 1.2)[4].

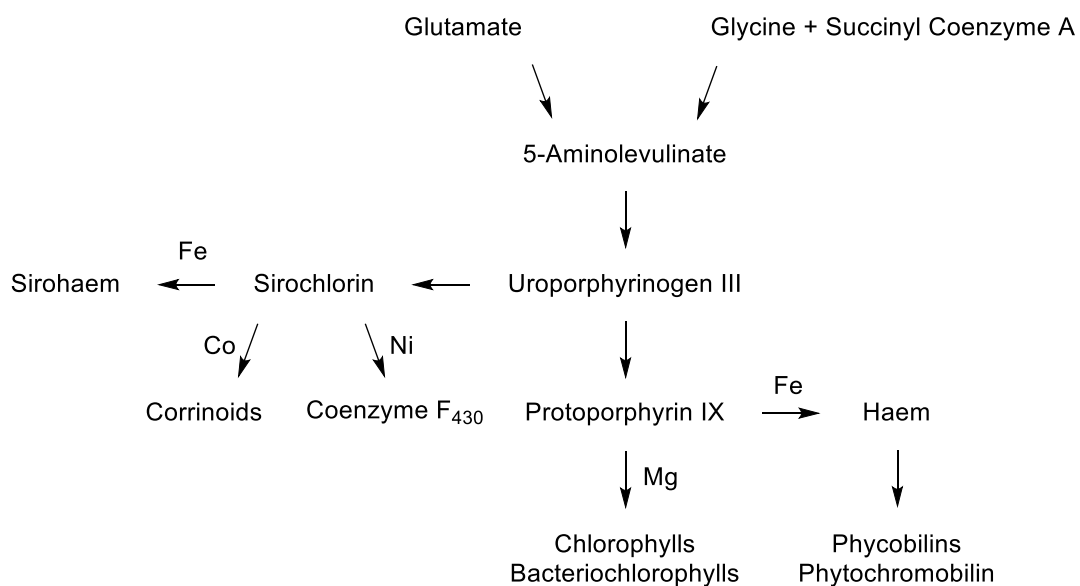


Figure 1.2. An outline of illustrative tetrapyrrole biosynthetic pathways

The biosynthesis of chlorophylls (Chl and Bchl) comprises a series of enzyme-catalysed steps which begins with the formation of 5-aminolevulinic acid (ALA). This can occur via one of two processes depending on the organism. The first branch point in the pathway occurs at the molecule uroporphyrinogen III, which can be used in any of the iron, magnesium, nickel, cobalt or alternative iron branches of metallo-tetrapyrrole synthesis. A second branch point occurs at protoporphyrin IX, the precursor molecule for both haems and chlorophylls (Figure 1.3). Magnesium chelatase operates at this branch point, catalysing the insertion of Mg^{2+} into the tetrapyrrole ring of P_{IX} in the first committed step towards chlorophyll biosynthesis. This reaction is likely to be a rate-controlling step in chlorophyll production[4–6].

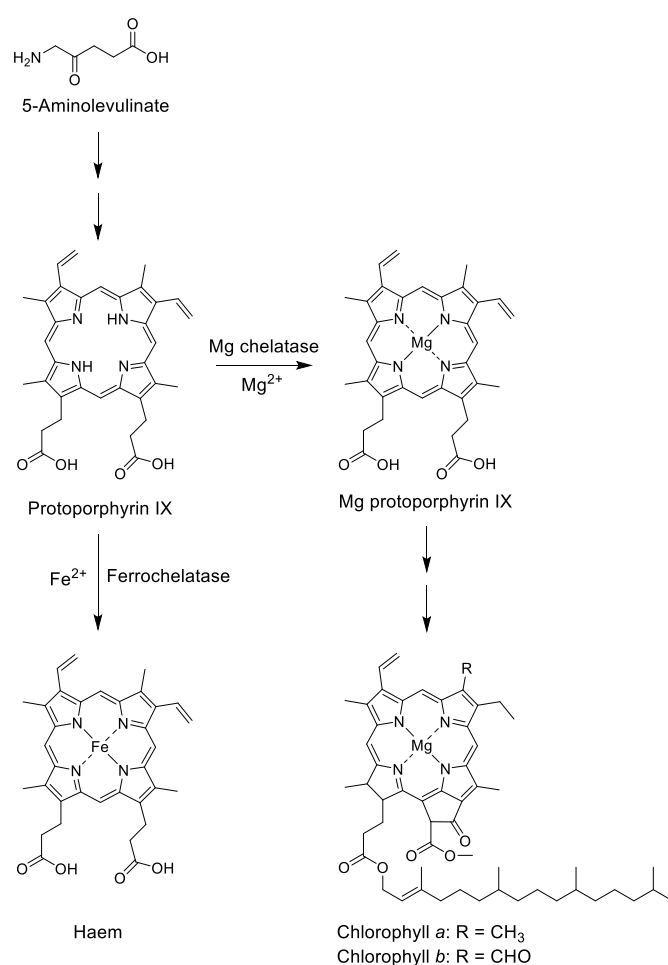


Figure 1.3. The branch point between the haem and chlorophyll biosynthetic pathways.

Magnesium chelatase catalyses the insertion of magnesium into protoporphyrin IX, the first committed step in the biosynthesis of chlorophyll.

The biosynthetic pathways of both chlorophylls and bacteriochlorophylls share common intermediates up to the synthesis of chlorophyllide. Because of the similarity between many of the steps of chlorophyll synthesis in plants, algae and bacteria, the study of photosynthetic processes in bacteria has aided understanding of the mechanism in higher plants. Several of the homologous enzymes involved were first discovered and characterized in photosynthetic bacteria[4,7].

1.1.2. Structure and function of the subunits of magnesium chelatase

The enzyme magnesium chelatase consists of three protein subunits: Bchl, BchD and BchH in organisms producing bacteriochlorophyll; and ChlI, ChlD and ChlH in those synthesizing chlorophyll. These will be referred to generically as I, D and H subunits in the following unless specifically citing the bacteriochlorophyll- or chlorophyll-associated form. Homologous subunits from different species show sequence similarity in their primary structures and are of similar size: I subunits range between 38 kDa and 42 kDa, D subunits range between 60 kDa and 74 kDa, and H subunits range between 140 kDa and 150 kDa[8]. Similarities in primary structure between I and D subunits suggest these proteins also share some functional features[9].

1.1.2.1. The I subunit

The I subunit has been identified as a member of the AAA⁺ (ATPases associated with various cellular activities) superfamily of enzymes and an ATPase through sequence analysis, 3D structural analysis, and ATPase assays[10–13]. It binds to the D subunit to form a multimeric structure[11,14,15]. It is also known to form homomeric ring structures although there is some debate as to the exact stoichiometry and functional relevance of these complexes which are most likely fragile and transient[12,16,17].

1.1.2.1.1. Primary and tertiary structure

Homologues of the I subunit share varying degrees of primary sequence similarity. ChlI protein from *Synechocystis* shows 51 % identity to Bchl from *Rhodobacter capsulatus*[18]. All homologues of the I subunit from bacteria, algae and plants have been shown to possess the Walker A (GX₄GK(T/S); X is any amino acid) and Walker B (hhhhhDE; h is any hydrophobic amino acid) nucleotide-binding motifs characteristic of members of the AAA⁺ protein superfamily near the N-terminus in their primary structures[9,10,19,20]. An integrin I domain-binding motif found in the primary structure of Bchl has been proposed to bind the integrin I domain identified in the primary structure of BchD during the functional cycle of magnesium chelatase[10].

The 3D structure of crystallized Bchl was solved to a resolution of 2.1 Å [10]. This confirmed it belongs to the AAA⁺ superfamily by comparison with other available structures of AAA⁺ proteins, although it displays a novel arrangement of domains (Figure 1.4). Its C-terminal domain consists of a bundle of four α -helices, but instead of sitting on top of the nucleotide binding site, this domain is located behind it. In this position the helical bundle of Bchl contacts the N-terminal domain over a larger area than is found in other members of the AAA⁺ superfamily with standard domain architecture[10]. The N-terminal domain containing the Walker A and B motifs is organized into a Rossmann fold (a structural motif of alternating β strands and α helices arranged to form a β sheet, characteristic of nucleotide-binding domains[20]). This fold is likely present in I subunit homologues from different species judging by the high level of sequence identity between them. Several hairpins protrude from the bulk of the N-terminal region and contribute to the formation of a deep, positively-charged groove. This groove is open to the surface of the molecule and might be involved in interactions with the other magnesium chelatase subunits.

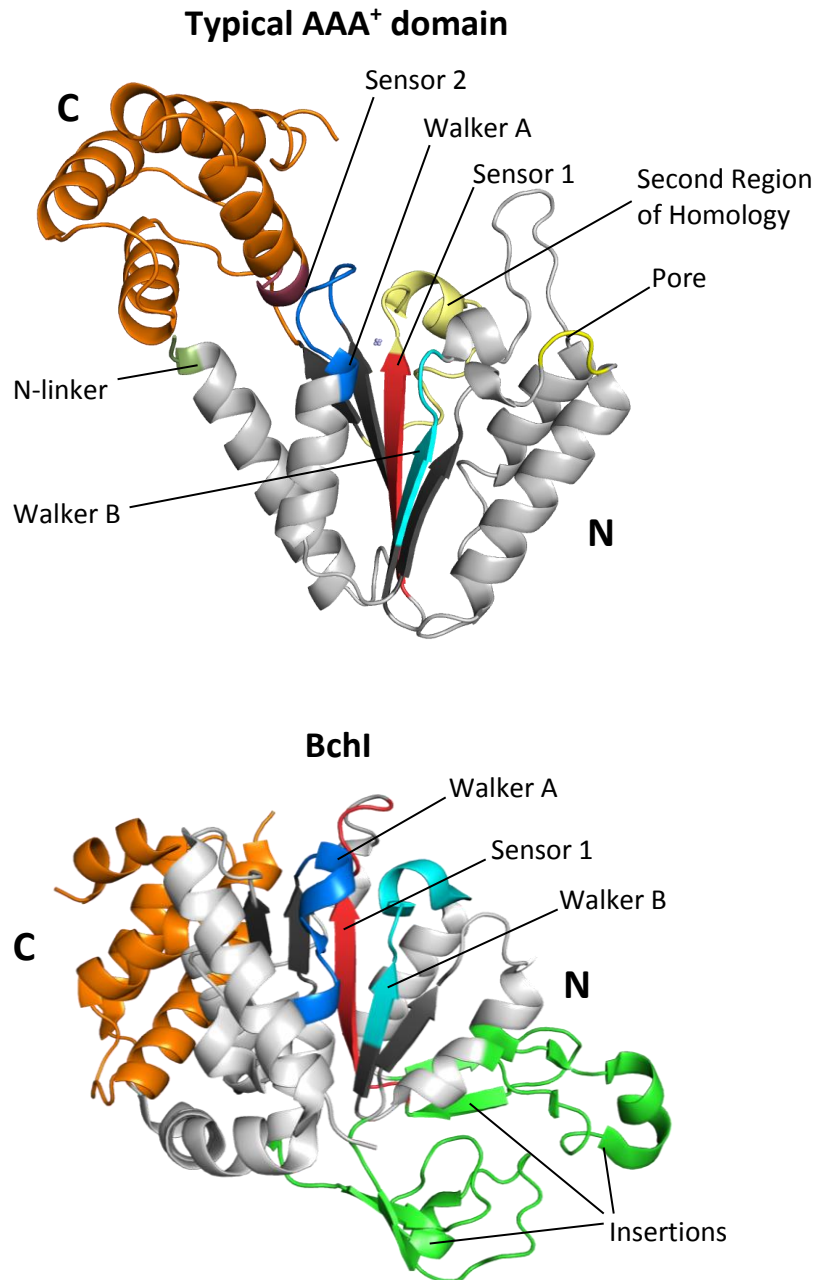


Figure 1.4. Structures of a typical AAA⁺ domain and Bchl.

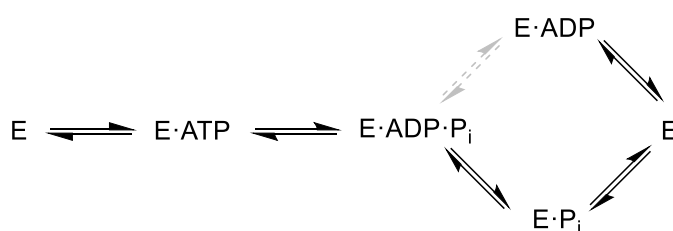
Cartoon representations of a typical AAA⁺ domain (NSF protein; PDB code 1D2N) and the Bchl subunit from *R. capsulatus* (PDB code 1G8P). The colours used indicate key elements common to Bchl and other members of the AAA⁺ superfamily. In a typical AAA⁺ domain the C-terminal domain (orange) is positioned so that it partially encloses the nucleotide binding site in the wedge-shaped N-terminal domain (grey). Bchl exhibits a novel arrangement of these domains where the C-terminal domain is located behind the nucleotide-binding site of the N-terminal domain. The insertions highlighted in green are specific to Bchl. They form hairpins which protrude from the main core of the subunit.

1.1.2.1.2. ATPase activity

Various homologues have been shown to possess ATPase activity[11,19,21]. ChII from *Synechocystis* was shown to catalyse the hydrolysis of MgATP²⁻ with a k_{cat} of 0.048 ± 0.003 s⁻¹ in the presence of 5 mM free Mg²⁺ [12]. Free Mg²⁺ ions were found to be activators of

the process, building on previous findings that ATP hydrolysis would only occur when MgCl_2 was in excess over ATP[11]. Transient kinetic studies using the fluorescent ATP analogue TNP-ATP revealed behaviour characteristic of a two-step mechanism where free ChII undergoes slow isomerization affected by Mg^{2+} , followed by fast ligand binding[12]. The binding kinetics changed markedly with Mg^{2+} concentration, suggesting Mg^{2+} was binding to all enzyme forms.

It is possible that the ATP hydrolysis reaction proceeds via two different pathways (Scheme 1.1). Evidence supporting the pathway containing an enzyme-phosphate complex in ChII from *Synechocystis* was found by nonequilibrium isotope exchange[22] following reports that the Bchl subunit catalyses a phosphate exchange reaction from ATP to ADP[21,23]. The same study demonstrated that the ATPase activity of the ChIID and ChIIDH complexes also followed this pathway. No evidence was observed for the pathway containing the enzyme-ADP complex for ID or IDH, but the ChII subunit on its own exhibited a very small amount of exchange which could not be quantified.



Scheme 1.1. Possible reaction pathways for ATP hydrolysis by magnesium chelatase.

The hydrolysis of ATP by the I subunit of magnesium chelatase proceeds either via an enzyme-phosphate or an enzyme-ADP complex. Experimental evidence supports the pathway containing an enzyme-phosphate complex[22], shown by solid arrows. The enzyme-ADP complex is suggested to act as a dead-end inhibitor.

1.1.2.1.3. Oligomeric structure

As members of the AAA⁺ superfamily, it was long supposed that I subunits of magnesium chelatase would associate into hexameric rings similar to those observed for other AAA⁺ proteins (Figure 1.5)[24]. Oligomeric complexes have been observed for both ChII[12] (heptameric rings from negative stain electron microscopy) and Bchl[10,16] (hexamers observed by negative stain electron microscopy), but it is not clear that these structures represent part of the active chelatase complex. Solution studies of recombinant *Synechocystis* ChII subunits found that their association was concentration dependent[12,17]. In the absence of nucleotide and metal ion, size exclusion chromatography revealed that the apparent molecular mass of ChII aggregates increased from 81 kDa to 208 kDa as the concentration of ChII was increased from 20 μM to 200 μM [17], while in the presence of MgATP^{2-} the elution volume was independent of protein concentration and indicated a high-molecular-mass aggregate of 6 to 8 subunits. At concentrations ranging from 1.1 μM to 56 μM , another SEC experiment found the majority of ChII eluted as a monomer in the absence of substrates[12]. The elution profile gradually shifted to higher mass aggregates with increasing protein concentration in the presence of

substrates. It has since been stated that the heptamer does not reflect the dominant species in solution[25].

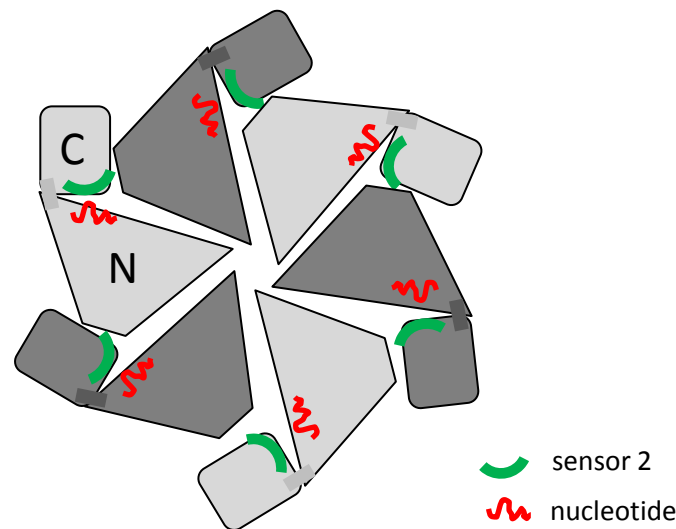


Figure 1.5. Oligomeric configuration of AAA⁺ domains.

The C- and N-terminal domains are indicated by C and N respectively. The AAA⁺ domains of typical family members come together in a ring structure to form oligomers. These oligomers generally comprise six protomers and represent the biologically active form. In this arrangement the C-terminal domain sits on top of the N-terminal domain so that the sensor 2 region of a subunit partially covers its own nucleotide binding site. The C- and N-terminal domains of one subunit are also in contact with the N-terminal domain of the adjacent subunit.

The stoichiometry of the I oligomer remains to be determined. Results from studies of the same protein differ, indicating at the same time that association of the subunits is concentration and substrate dependent[12,17,25]. While Jensen et al. observed via gel filtration that ChII subunit association was concentration dependent in the absence of substrates[17], Reid et al. found that ChII eluted as a monomer in the absence of substrates regardless of protein concentration[12]. This discrepancy might be due to the fact that the former study employed His-tagged protein. Although histidine tags are widely used to simplify and expedite protein purification without adverse effect on the biochemical properties of the recombinant protein, there have been reports of histidine tags promoting or stabilizing dimerization and oligomerization not observed in the wild type[33,34]. Nevertheless, the I subunit is capable of forming various homo-oligomeric structures but it is unclear which of them, if any, are associated with the active chelatase complex. In the presence of MgATP²⁻ Reid et al. observed the molecular mass of eluted ChII aggregate to increase gradually from monomer to hexamer with increasing protein concentration[12]. Asymmetric peaks in the SEC trace suggested the complexes were unstable. AUC of fluorescein-labelled ChII supports this view by revealing the existence of a range of different complexes in the presence of substrates[25]. The fact that preincubating His-ChII with substrates before addition to a chelatase assay considerably increases the lag time before the start of product formation indicates that the preformed ChII hexamer is not on

the catalytic pathway[17]. Furthermore, attempts to determine the state of the I subunit oligomerization[10,12,16,17] achieve oligomers at protein concentrations which are often 10 times that used in chelatase assays[26,28–30].

1.1.2.2. The D subunit

The D subunit has also been identified as a member of the AAA⁺ superfamily through sequence homology[10,21], although it only possesses a degenerate form of the ATP-binding motif found in the I subunit and has not been observed to hydrolyse ATP[11,21,23]. It associates with the I subunit in the presence of nucleotide[11,14,15] and allosterically regulates substrate binding to the chelatase complex[30].

1.1.2.2.1. Primary structure

The primary structure of ChlD from *Synechocystis* is divided into two major domains separated by a proline-rich region[9]. Its N-terminal domain shares 40 % to 41 % identity with *Rhodobacter* Bchl and *Synechocystis* ChII but only 18 % identity with the same region of *Rhodobacter* BchD while its C-terminal domain shows 33 % identity with BchD. The integrin I domain located at the C terminus is proposed to bind to the integrin I domain-binding motifs of I and H during the functional cycle of magnesium chelatase[10].

The N-terminal domain encompasses the AAA module, and in some homologues this contains a degenerate form of the Walker A motif found in the I subunit[19,21]. Neither BchD nor ChlD possesses detectable ATPase activity[11,21]. Chelatase assays performed with ChlD subunits in which key residues of the AAA⁺ domain were mutated have shown the importance of the AAA module in ChlD for the chelatase complex[30]. It was found that the mutations caused a significant decrease in chelatase activity and affected the specificity of the complex for its substrates.

1.1.2.2.2. Bchl is required for correct folding of BchD

When expressed in *E. coli*, recombinant BchD is largely insoluble and must be purified under denaturing conditions[18,19,35]. Active BchD was then obtained by rapid dilution or dialysis. The yield of active BchD was increased by dilution in the presence of Bchl, suggesting this subunit is required for correct folding of BchD. Lake et al. confirmed this suggestion, demonstrating that Bchl acts as a chaperon essential for the survival of BchD *in vivo*[15]. This is in contrast to the ChlD subunit from *Synechocystis* which is soluble when overexpressed in *E. coli*[9].

1.1.2.3. The ID complex

It was proposed that the I and D subunits come together in a complex during the initial phase of the magnesium chelatase reaction due to the fact that preincubating these subunits with ATP and Mg reduced the lag phase observed at the beginning of the reaction[17,18,36]. This association has been shown directly via pull down assays in which ATP and Mg were found to be absolute requirements for complex formation[11,19]. Nucleotide hydrolysis is not necessary since ATP was successfully substituted with ADP as well as the slowly-hydrolysable analogue adenosine 5'-[γ-thio]triphosphate and the non-hydrolysable analogue adenosine 5'-[β,γ-imido]triphosphate after complex formation[11]. SDS-PAGE analysis of the eluted ID complexes suggested ChII and ChlD associate in a 1:1 ratio.

The structure of the ID complex has proven much more elusive. Single particle reconstructions from cryo-EM images of the ID complex from *R. capsulatus* produced a double ring structure where 6 BchD subunits and 6 Bchl subunits formed homomeric hexamers stacked one on top of the other[14,37]. This is the only ID homologue which has been successfully isolated and subjected to high resolution structural analysis.

The ID complex exhibits ATPase activity, although this is probably due to the ATPase site in the I subunit. The level of ATPase activity of ID complexes from *R. sphaeroides* is virtually identical to that of Bchl on its own[19]. In *Synechocystis* the ATPase activity of ChII is repressed when ChII and ChID are assayed together, suggesting that ChID inhibits the activity of ChII[11].

1.1.2.4. The H Subunit

The H subunit binds the protoporphyrin substrate and so probably contains the chelatase active site[28,31,38]. ChIH does not exhibit ATPase activity[11], an observation later confirmed in BchH despite earlier incorrect reports[27]. One view of the chelatase reaction, drawing on the AAA⁺ literature, is that the H-porphyrin complex forms a macromolecular substrate for the ID AAA⁺ complex.

1.1.2.4.1. Primary and tertiary structure

Single particle reconstruction of negative stain electron micrographs (resolution 25-28 Å) reveals the BchH protein to be ~100 Å in diameter with a depth of 35-75 Å, and consisting of three major lobes, I, II and III, approximately equal in size (Figure 1.6 C)[28]. Lobe I possesses an additional 'thumb'-shaped density protruding from its tip, and Lobe III has a poorly defined 'finger'-shaped density. The poor definition of the 'finger' indicates its flexibility since the reconstruction contains an average of a range of positions it can occupy.

Protease digestion was employed to probe the flexibility and domain organisation of the protein since proteolysis tends to occur at exposed flexible sites of the polypeptide chain[28]. 16 h of digestion resulted in a major fragment of ~45 kDa remaining. This proteolysis-resistant fragment was identified as the C-terminal domain by mass spectrometry, encompassing residues 734-1194. Binding of protoporphyrin was found to stabilise BchH during the 16 h digestion so it remained at its native size. This suggests a substrate-induced conformational change of sufficient magnitude to increase structural rigidity and impede proteolytic cleavage.

Truncated N- and C-terminal domains of BchH (BchH-Ndom residues 1-733, BchH-Cdom residues 734-1194 respectively) do not bind deuteroporphyrin IX (D_{IX}; a more soluble substrate analogue of P_{IX}), suggesting that residues from both domains form the active site[28]. EM images of BchH bound to protoporphyrin IX clearly reveal a structural change in which the finger and thumb domains come into direct contact, representing a movement of ~30 Å (Figure 1.6D)). Further characterization of BchH-Ndom and BchH-Cdom reveals that the N-terminal domain comprises lobe I, lobe II and the thumb regions while the C-terminal domain contains lobe III and the finger regions. This supports the hypothesis that porphyrin binding induces interaction between the two domains. It is not clear, however, where the porphyrin-binding site is on the protein; the substrate could be clamped

between the finger and thumb, or it could be bound deeper in the cleft, closer to the hinge region between the N- and C-terminal domains.

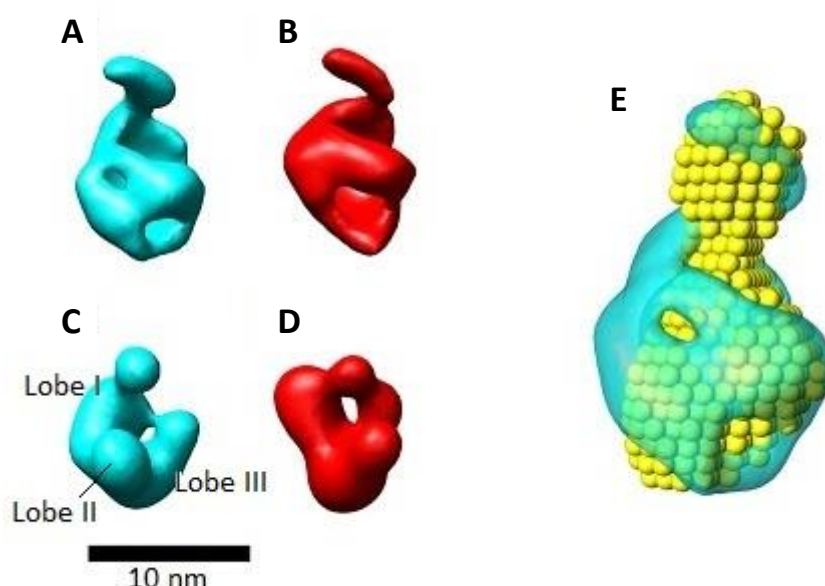


Figure 1.6. 3D models of ChlH and BchH with and without protoporphyrin IX substrate bound.

3D models calculated at a cut-off resolution of 30 Å of A) apo-ChlH, B) ChlH-D_{IX} complex, C) apo-BchH and D) BchH-protoporphyrin IX complex. E) Superposition of *Thermosynechococcus elongatus* apo-ChlH model from EM analysis (cyan) with *Synechocystis* apo-ChlH model from SAXS (yellow)[38]. These models were originally published in the Journal of Biological Chemistry. Qian, P. et al. Structure of the cyanobacterial Magnesium Chelatase H subunit determined by single particle reconstruction and small-angle X-ray scattering. J. Biol. Chem. 2012; 287:4946-4956. © the American Society for Biochemistry and Molecular Biology.

The structures of *Synechocystis* and *Thermosynechococcus elongatus* ChlH subunits have been analysed via small-angle X-ray scattering and negative stain EM single particle analysis respectively[38]. No structural differences between free and deuteroporphyrin IX-bound protein (apo-ChlH and ChlH-D_{IX}) from *T. elongatus* were observed at 30 Å resolution (Figure 1.6 A and B). This ChlH has a hollow structure consisting of an internal cavity of ~100 nm³ which is 3 to 4 times the size of a P_{IX} molecule, and a small globular domain attached to the rest of the protein by a narrow neck. The globular ‘head’ region was identified as the N-terminal domain by attaching nanogold particles to the N-terminal His₆ tag on the protein. Estimates of the fraction of ChlH represented by the head domain place its molecular mass in the range 14.1 kDa to 17.3 kDa.

The lower resolution structure obtained for *Synechocystis* ChlH (a 3D matrix of hexagonally distributed atoms spaced by 9 Å reconstructed from SAXS) reveals a two-domain structure with dimensions in good agreement with the single particle reconstruction of *T. elongatus* ChlH when the two structures are superimposed (Figure 1.6 E). Likewise, no difference can be observed between the apo-ChlH and ChlH-D_{IX} complexes. By deleting 63.2 kDa (566 residues) of the N-terminal region of *Synechocystis* ChlH in an attempt to disrupt the ‘cage’ structure of ChlH, it was found that the subunit’s porphyrin-binding ability was unaffected

but magnesium chelatase activity was completely abolished. This shows that porphyrin binding places fewer demands on the structure of ChlH than its catalytic function which requires its cage structure to be fully closed.

1.1.2.4.2. Porphyrin binding

When BchH from *R. sphaeroides* or *R. capsulatus* is expressed in *E. coli*, it is copurified with protoporphyrin IX, indicating the H subunit binds this substrate[6,18]. Deuteroporphyrin IX is also a substrate for H subunits from *Synechocystis* and *R. sphaeroides*[31]. Deuteroporphyrin IX is a more water-soluble alternative to the biological substrate protoporphyrin IX which will only dissolve in aqueous solution in the presence of detergent. Deuteroporphyrin binding follows a single-site binding model with a K_d of $(1.22 \pm 0.42) \mu\text{M}$ (determined by quantitative analysis of tryptophan residue quenching) for ChlH, and $(0.53 \pm 0.12) \mu\text{M}$ for BchH[31]. The H subunit has been shown by analytical HPLC gel filtration to form a monomeric complex when incubated in a molar excess of deuteroporphyrin, independent of the presence of Mg^{2+} or ATP. The presence of Mg^{2+} and ATP lowers the affinity of the H subunit for deuteroporphyrin, causing the K_d value to increase fourfold. Mg-deuteroporphyrin binds to the H subunit with a K_d similar to that for deuteroporphyrin. In chelatase assays deuteroporphyrin, unlike protoporphyrin, does not cause inhibition at high substrate concentrations ($>1 \mu\text{M}$)[17,31].

1.1.3. The magnesium chelatase reaction

Magnesium chelatase activity requires all three protein subunits, ATP and Mg^{2+} . The reaction consists of two phases. During the first phase, the I and D subunits form a complex dependent on protein and ATP concentrations. During the second phase this complex interacts with the H subunit to insert a magnesium ion into protoporphyrin IX with concomitant ATP hydrolysis. The H subunit acts as a substrate in the reaction with a K_m in the low micromolar range[39]. There are at least two requirements for ATP in the reaction: for the I-D interaction, and as an energy source for metal ion insertion[17].

1.1.3.1. Optimum subunit proportions

Optimum *R. sphaeroides* subunit proportions for magnesium chelatase activity have been investigated through titration experiments[19]. One protein was present at saturating concentration, another at limiting concentration, while the concentration of the third was varied. The results were ambiguous, however, since the assays where BchH was saturating and either Bchl or BchD was varied provided different optimum ratios of Bchl to BchD. It was found that when *Synechocystis* ChlH was added to limiting concentrations of ChlD and ChlI (corresponding to a ratio of D:H of 1:4), the increase in magnesium chelatase activity was linear up to proportions of I:D:H of 2:1:4 before saturation[11]. Work on determining optimum subunit proportions remains problematic, and as yet no consensus has been reached. The structures and proportions in which the subunits combine are very sensitive to experimental conditions.

1.1.3.2. Substrate requirements and kinetic analyses

Free Mg^{2+} is required for the reaction[19]. Essentially all ATPases use MgATP^{2-} as the true substrate, however magnesium chelatase is inactive unless an excess of MgCl_2 over ATP is present providing both MgATP^{2-} and Mg^{2+} [20]. Magnesium chelatase subunits from

Synechocystis exhibit a sigmoidal reaction rate dependence on Mg^{2+} concentration, indicating a cooperative response with $K_{0.5}$ of 3.8 mM[29].

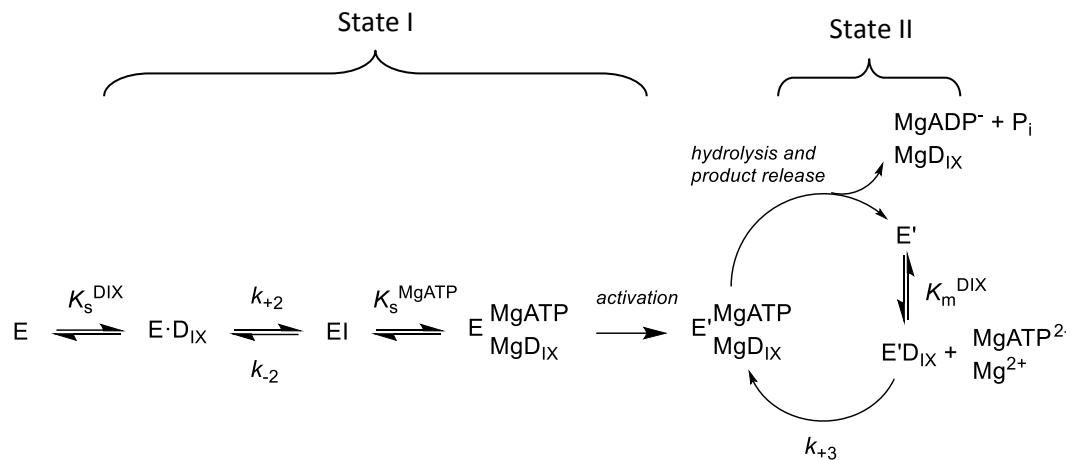
ATP hydrolysis was shown via a coupled ATPase assay to be an absolute requirement for the chelation step of the reaction[11]. The maximum rate of ATP hydrolysis coincided with the maximum rate of Mg^{2+} insertion, and was found to depend on the presence and concentration of the protoporphyrin substrate as well as the concentration of the H subunit. Results from magnesium chelatase assays with NEM-treated ChII (NEM is a thiol-modifying agent), where magnesium chelation did not proceed, suggest the process is driven by the ATPase activity of the I subunit[32].

The magnesium chelatase reaction of *Synechocystis* requires hydrolysis of $\sim 15 \text{ MgATP}^{2-}$ (deduced from the ratio of ADP or P_i to metalloprophyrin)[29]. The chelation partial reaction is energetically unfavourable under the assay conditions *in vitro*. This results in the chelatase reaction operating far from equilibrium *in vivo* given the likely metabolite concentrations. The steady-state kinetic behaviour of the intact enzyme was determined with K_{DIX} equal to $3.20 \mu\text{M}$, and a $K_{\text{MgATP}^{2-}}$ of 0.45 mM . k_{cat} for chelation is estimated to be 0.8 min^{-1} , suggesting ATP hydrolysis is catalysed by the isolated ChII subunit at a substantially lower rate than in the intact chelatase.

The steady-state rates of both ATP hydrolysis and metal ion chelation show similar, apparently cooperative, responses to free Mg^{2+} [29]. The response of magnesium chelatase to free Mg^{2+} was investigated further under saturating and limiting conditions of MgATP^{2-} and D_{IX} . When MgATP^{2-} and D_{IX} are saturating, a cooperative response is observed. While low concentrations ($\ll K_{\text{M}}$) of either MgATP^{2-} or D_{IX} separately do not greatly affect the cooperative response, it is not possible to determine whether the response at low concentrations of both substrates is cooperative or not, due to the exceptionally weak magnesium binding under these conditions. This indicates Mg^{2+} cooperativity is largely independent of substrate concentration.

Based on transient-state kinetic studies of the enzyme, a model of the magnesium chelatase reaction where the Mg^{2+} -saturated enzyme exists in one of two states separated by an activation step has been proposed[8]. Metal ion chelation was directly observed fluorometrically with enzyme in large excess over the porphyrin substrate to allow observation of single turnovers. The rate of formation of the enzyme-product complex $\text{E} \cdot \text{MgD}_{\text{IX}}$ was found to depend differently on the concentrations of enzyme (a hyperbolic dependence on ChIH concentration), MgATP^{2-} (an inversely proportional relationship indicating a two-step process where the first step is slow and the second fast), and Mg^{2+} (a sigmoidal trend indicating cooperativity). The transient-state kinetic data imply that the slowest first-order step leading to formation of an enzyme-product complex has a rate constant of $\sim 2.6 \times 10^{-3} \text{ s}^{-1}$, approximately five times slower than k_{cat} determined from steady-state data ($13 \times 10^{-3} \text{ s}^{-1}$). A model was proposed (applying to Mg^{2+} -saturated forms of the enzyme) which reconciled the different steady-state and transient values (Scheme 1.2): the first turnover promotes the enzyme to a more active form, the activation process being fast and irreversible. If the transient state kinetic parameters are assigned to the first,

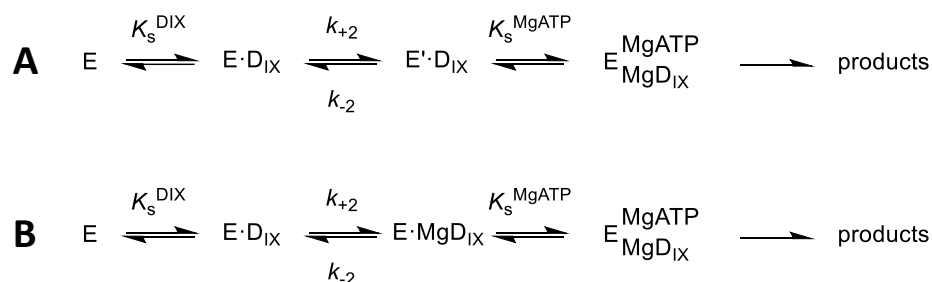
slower, turnover, the value found for k_{cat} is identical to that deduced from steady-state kinetic assays.



Scheme 1.2. Reaction scheme for action of magnesium chelatase in the presence of saturating Mg^{2+} .

The reaction consists of two phases. On completing the first slow turnover the enzyme is converted irreversibly to a more active form (from State I to State II) and subsequent chelation cycles are completed more quickly. The intermediate EI complex could take one of two forms (see Scheme 1.3).

Two models consistent with the data differ in the nature of the intermediate EI complex[8] (Scheme 1.3). One involves an enzyme-substrate complex which is primed for metal ion chelation, the other an enzyme-product complex. In both cases MgATP^{2-} then binds to the complex, shifting the equilibrium from the enzyme-bound porphyrin substrate towards metallo-porphyrin products.



Scheme 1.3. Possible reaction pathways for metal ion chelation.

The slow step is either A) the isomerization of D_{IX} -bound enzyme or B) metal ion chelation, followed by fast MgATP^{2-} binding and product release.

1.1.4. Summary

Magnesium chelatase is a complex enzyme comprising three different subunits which together form the active complex. The H subunit contains the porphyrin binding site and acts as a substrate to the ID complex. The I subunit is an ATPase, hydrolysing ATP to fuel the chelation reaction. It associates in an ATP-dependent manner with the D subunit which performs an allosteric regulatory role within the complex. The complex formed between

the I and D subunits is not well characterized. Titration experiments assaying optimum subunit concentrations have shown varying and inconclusive results. Unravelling the exact form of the active ID complex will provide much-needed insight into the complex, multi-step chelation reaction, as will a more detailed understanding of the ATPase activity of the I subunit.

1.2. Single molecule microscopy

1.2.1. Fluorescence microscopy

Fluorescence microscopy is an important tool in molecular and cell biology for non-invasive (reducing the perturbation of the system under study), time-resolved imaging. Through the use of fluorescent molecules (either intrinsic to the specimen under study, or deliberately added), objects with dimensions below the diffraction limit of visible light can be resolved[40] and even tracked over time[41,42]. The specificity afforded by fluorescent labelling enables different parts of a cell, or different proteins, to be distinguished from one another and imaged simultaneously[43,44]. Fluorescent labels can also be used to report on the immediate environment of a molecule as their fluorescence emission is often highly sensitive to factors such as pH and viscosity[45,46]. With the advent of diffraction-limit-breaking super resolution microscopy techniques and improvements in imaging technologies, fluorescence microscopy can now access the realm of the single molecule with relative ease.

1.2.1.1. The phenomenon of fluorescence

An atom or molecule is fluorescent when it is capable of absorbing energy in the form of light at a certain wavelength and re-emitting the energy in the form of light at a different wavelength. Generally the emitted light will have a longer wavelength (less energy) due to the excited molecule dissipating energy through interactions with its environment before it emits its fluorescence, a circumstance known as Stokes' shift. This phenomenon of fluorescence requires a source of energy external to the molecule in the form of electromagnetic radiation[47].

A molecule possesses quantized energy levels which arise from three of its properties: the arrangement of its electrons (electronic orbital configuration), the motions of its nuclei (vibrations and rotations), and the intrinsic angular momentum of its electrons (spin). Figure 1.7A illustrates a typical arrangement of the lower electronic energy states in a molecule (the ground state S_0 , the lowest singlet excited state S_1 , and the lowest triplet state T_1), known as a state energy diagram[48]. Singlet and triplet states emerge from pairing of electron spins in accordance with the Pauli exclusion principle. Within each electronic state there exist multiple vibrational energy levels, illustrated for S_1 in Figure 1.7B[49].

Upon absorption of a photon with sufficient energy, a fluorescent molecule (fluorophore) will become excited as an electron in its ground state uses the energy to transition to a higher-energy excited state (Figure 1.7A). The absorption and excitation process occurs on a timescale of 10^{-15} s. Once excited, the fluorophore will most likely relax to the lowest singlet excited state (S_1) through non-radiative processes within 10^{-13} - 10^{-11} s. It can remain

in S_1 for periods between 10^{-9} s and 10^{-7} s before relaxing to the ground state with emission of a photon (fluorescence).

In order to understand the structure observed in fluorescence excitation and emission spectra, it is necessary to consider the quantum mechanical description of the vibrational energy sublevels (Figure 1.7B). The electron is assumed to occupy the lowest vibrational sublevel of the ground state ($v = 0$) since this is usually the most populated, and the energy of the absorbed photon (wavelength) determines which vibrational energy level the electron transitions to within an excited state. The probability that a photon of a particular energy will be absorbed depends on the degree of overlap between the wavefunctions of the vibrational sublevels involved (the Franck-Condon principle). A vertical transition is more likely to occur between vibrational sublevels for which the wavefunctions overlap maximally, as is the case for the solid arrow indicating a transition between the $v = 0$ level of S_0 and the $v = 3$ level of S_1 in Figure 1.7B. Other transitions are possible between S_0 and S_1 , but are less probable because the wavefunction overlap is not as great (compare the transition indicated by the light dashed arrow, where the $v = 1$ wavefunction of S_1 is maximally positive, but that of the $v = 0$ ground state is weakly positive). Emission of a photon follows the same principle, except the sublevels involved are the $v = 0$ level of S_1 and any of the vibrational sublevels of S_0 . The relative probabilities of photon absorption and emission translate into the intensities observed at different wavelengths of light in excitation and emission spectra. Individual molecular vibrational energy levels cannot usually be resolved, but merge to form a band. The bands are further broadened when a molecule is in solution due to coupling of the fluorophore molecule with solvent molecules which produces many different supramolecular configurations, each with a slightly different energy (Figure 1.8).

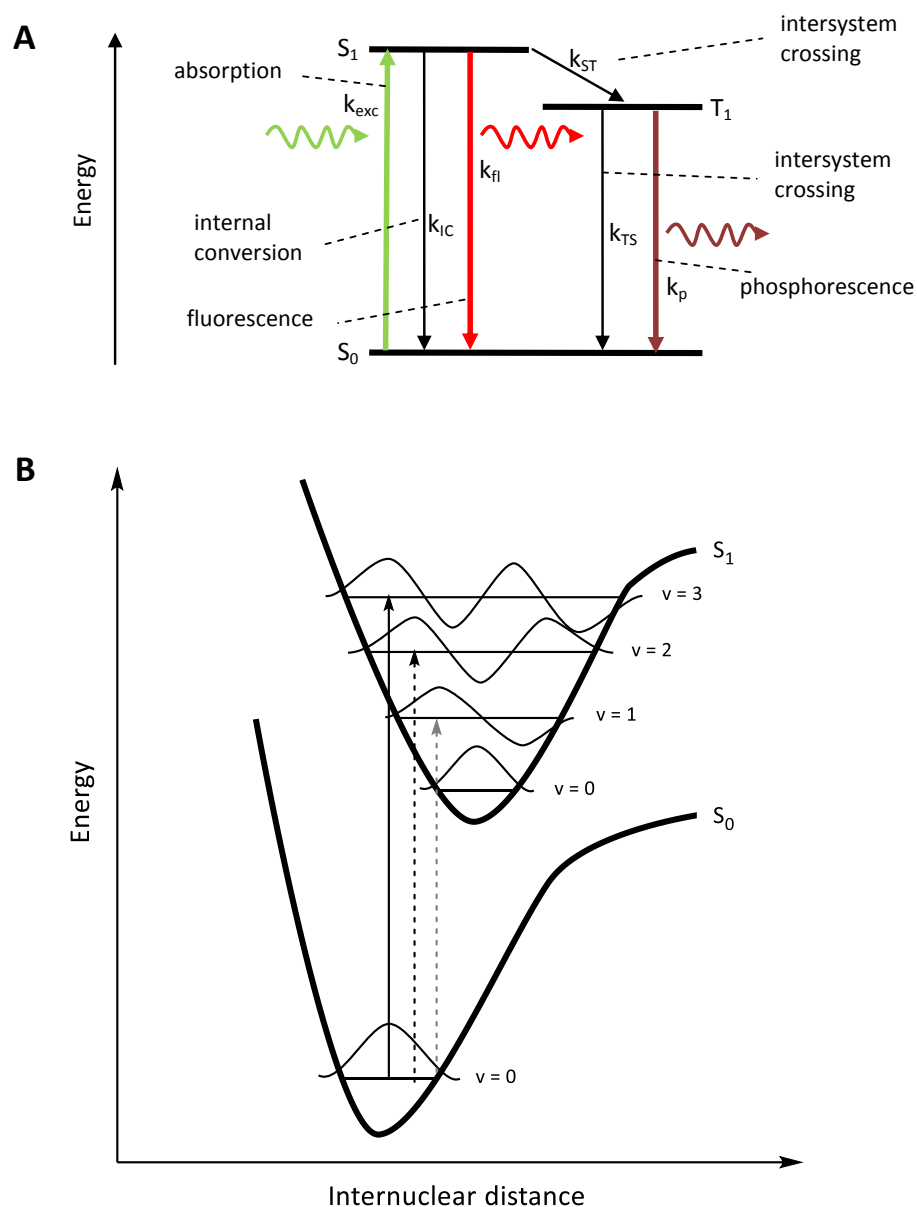


Figure 1.7. Electronic energy level diagrams illustrating photophysical processes.

A) State energy diagram illustrating the ground (S_0), lowest excited singlet (S_1) and lowest triplet (T_1) electronic states. The vertical axis represents the energy of the system while the horizontal axis has no physical meaning. Transitions between the states are shown with arrows; all radiative processes are indicated by a coloured arrow with accompanying photon (horizontal squiggly arrow). B) Quantum mechanical anharmonic oscillator model of electronic and vibrational energy levels showing the energy of the system as a function of internuclear separation. Each electronic energy level is associated with a particular electron arrangement and corresponding equilibrium nuclear arrangement. The frequency of oscillation of the nuclei determines the vibrational energy sublevel ($v = 0, 1, 2, \dots$). The vibrational wave functions are shown for each of the sublevels. Electronic transitions resulting from absorption of a photon are indicated by arrows from most probable (solid arrow) to least probable (light dashed arrow) according to the Franck-Condon principle.

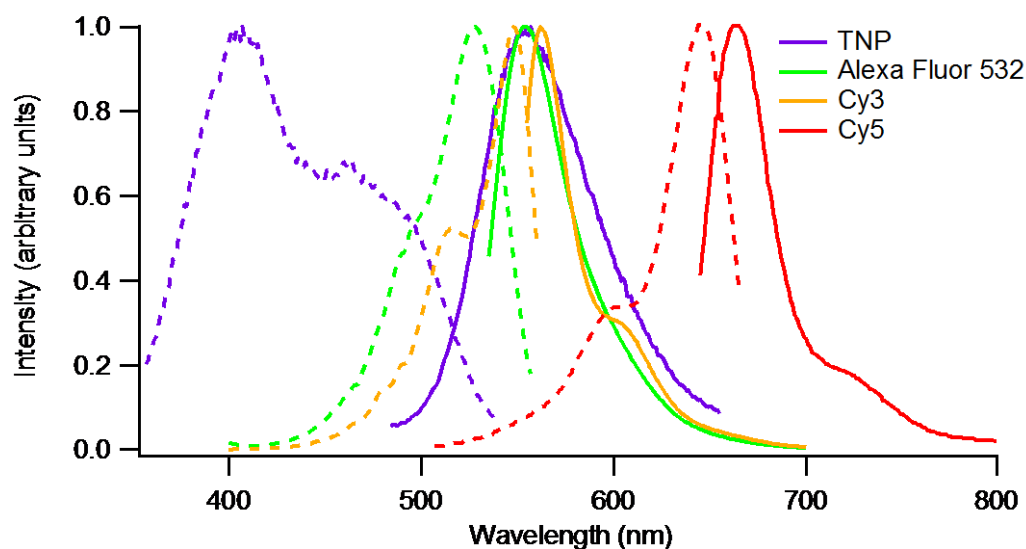


Figure 1.8. Fluorescence excitation and emission spectra.

Excitation (dashed line) and emission (solid line) spectra for the organic dyes TNP, Alexa Fluor 532, Cy3 and Cy5.

Under continuous illumination a fluorophore can undergo many cycles of excitation and emission before permanent photobleaching occurs. For example, a single molecule of fluorescein isothiocyanate (FITC) can emit approximately 30 000 fluorescence photons before bleaching[47,48].

The non-radiative processes by which a molecule loses energy and relaxes to the lowest vibrational sublevel of S_1 are internal conversion and vibrational relaxation. Vibrational relaxation is the transition of the electron from higher vibrational energy levels to lower ones within the same electronic energy state by loss of kinetic energy to other vibrational modes within the molecule, or to surrounding solvent molecules. Internal conversion is very similar to vibrational relaxation, except it occurs between vibrational energy levels in different electronic states. This is more likely to happen amongst the higher excited states where the gaps between vibrational energy levels becomes ever smaller and vibronic levels from one electronic state overlap those of another electronic state. The fact that there is a large gap between S_0 and S_1 , and no overlap of vibrational and electronic energy states, means that other transitive processes can compete with internal conversion for relaxation back to the ground state, namely fluorescence, intersystem crossing and quenching.

Intersystem crossing is normally a relatively rare event compared to fluorescence, occurring on timescales as long as 10^{-3} s. It is the transition of the electron as it changes spin multiplicity, for example from $S_1 \rightarrow T_1$ or $T_1 \rightarrow S_0$. Once in the excited triplet state, the electron can undergo vibrational relaxation to the lowest triplet vibrational sublevel and return to S_0 via phosphorescence with the emission of a photon. This is another slow process occurring on timescales between 10^{-4} s and 10^{-1} s. Alternatively, intersystem crossing can result in delayed fluorescence when the electron crosses back to S_1 .

Quenching is the final non-radiative process to consider which competes with fluorescence. Various types of quenching can occur, all of which result in the return of the fluorophore to

the ground state without fluorescence emission. In collision or dynamic quenching, the excited fluorophore collides with another molecule, transferring its excited state energy in the process. In static or complex quenching, fluorophores and quencher molecules form complexes which reduce absorption of excitation energy by depleting the population of fluorophores capable of being excited. In resonance energy transfer (RET), an excited fluorophore transfers its energy to an acceptor molecule which has an absorption spectrum overlapping the emission spectrum of the fluorophore.

1.2.1.2. The fluorescence microscope

The fluorescence microscope was developed in the early part of the 20th century. It is designed to deliver excitation light to a fluorescent sample and return the much weaker fluorescence emission, separated from the excitation illumination. This is possible thanks to the phenomenon of Stokes' shift whereby fluorescence is emitted at longer wavelengths than the excitation illumination, so the two can be separated by the use of dichroic mirrors and filters[47].

The most commonly used illumination arrangement is called incident-light or epi-illumination where the objective lens functions as both the condenser and the objective. Figure 1.9 illustrates the principle of operation of a typical inverted fluorescence microscope using epi-illumination. Excitation light is directed into the objective lens by a dichroic beam-splitting mirror. This mirror is designed to reflect light of shorter wavelengths but transmit light of longer wavelengths. The objective lens focusses the excitation light onto the sample and simultaneously collects the fluorescence emission. It directs the light back to the dichroic mirror which allows the fluorescent light to pass through. An emission filter cleans up the fluorescence emission by only allowing light with wavelengths in a narrow band to pass through, so that any stray excitation light or luminescence arising from components of the objective lens are removed.

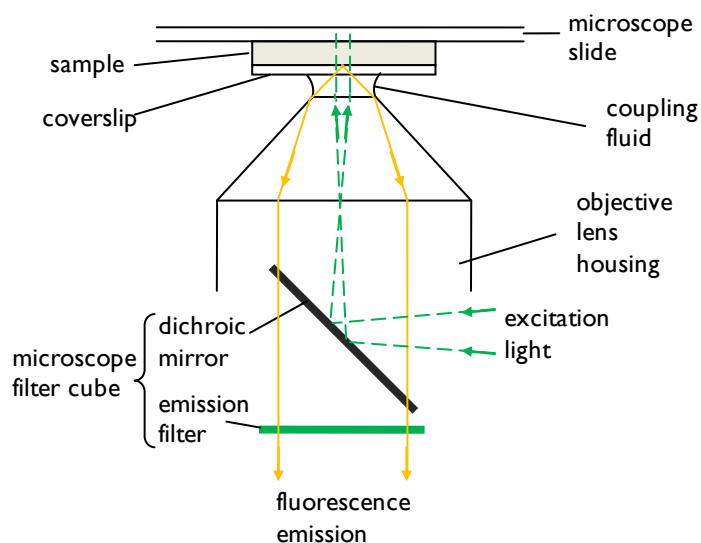


Figure 1.9. Principle of operation of an inverted fluorescence microscope using epi-illumination.

Excitation illumination entering the microscope objective is reflected towards the fluorescently-labelled sample by a dichroic mirror. The dichroic mirror is selected to reflect light with wavelengths in the region of the excitation illumination but transmit light of longer wavelengths so that the fluorescence emission from the sample passes through to the eyepiece or a detector for viewing. An emission filter prevents stray light of other wavelengths from interfering with the fluorescence image.

The excitation light can come from a white light source, in which case an excitation filter is used to select the desired wavelength band, or from a laser. Lasers provide a powerful source of collimated, monochromatic illumination. The coupling fluid used depends on the objective lens, ranging from air to water, oil or glycerol for example. It serves to match the refractive indices of the objective lens and the sample coverslip to reduce unwanted reflections. The choice of coupling fluid also determines the refractive index of the medium in direct contact with the objective lens, and hence controls the efficiency with which the lens gathers light. The filtered fluorescence emission is transmitted to either the eyepiece for direct viewing, or to a detector such as a photomultiplier tube or camera coupled to a recording device.

1.2.1.3. The diffraction limit and resolution

The resolution of an imaging system is defined as the smallest distance two objects can be separated by whilst still being discernible as two individual objects. Even in an aberration-free optical system, there is a fundamental limit dictated by the wavelength of the radiation used for imaging and the size of the objective lens pupil beyond which the system cannot resolve details of the object under observation. This is due to the wave-like nature of light which causes it to diffract and spread out when passing through a gap of comparable size to its wavelength[47,50].

Consider a point source of light (a good approximation of a single fluorescent molecule) viewed under a microscope. The light emitted from the point source will be diffracted on passing through the pupil of the objective lens so that the point appears as a spot of finite dimensions. The diffraction pattern produced by light passing through a circular aperture is

in fact an Airy pattern, a series of concentric rings around a central disc, but the Airy disc is so bright compared to the rings that they can be disregarded. The radius of the point source image, r_{Airy} , is determined by the Rayleigh limit to depend on the wavelength of the emitted light, λ , and the numerical aperture of the objective, NA, according to the relationship[47]

$$r_{\text{Airy}} = 1.22 \frac{\lambda}{2 \cdot NA} \quad \text{where} \quad NA = n \sin \theta_{\text{max}}$$

Equation 1.1. Calculation of the radius of a point source image

The numerical aperture is a dimensionless number characterizing the range of angles over which the objective can gather light: the higher the NA, the greater the angular spread over which light is accepted. Its value depends on the refractive index of the immersing medium next to the objective (n), and the half-angle of the maximum cone of light accepted by the objective (θ_{max}). The point source image is also known as the point spread function (PSF) of the imaging system. The Rayleigh criterion states that for two such point sources emitting light of the same wavelength and separated by a centre-to-centre distance, d , their images will no longer be distinguishable when $d \leq r_{\text{Airy}}$ (Figure 1.10).

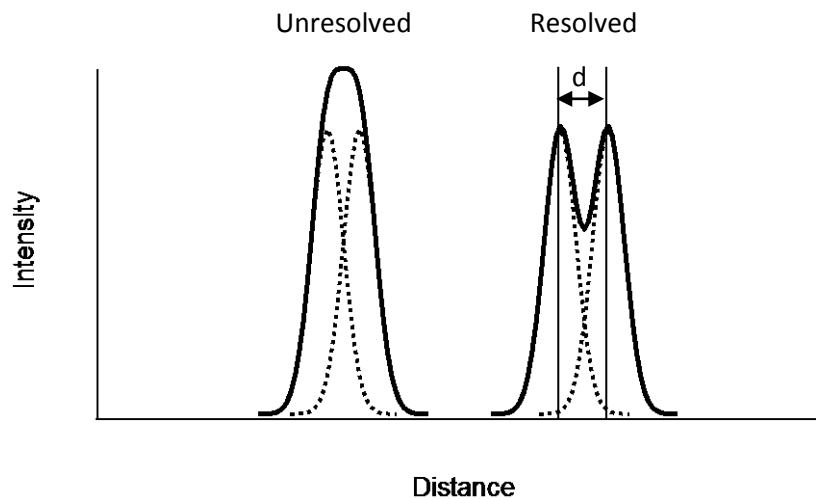


Figure 1.10. Distinguishing two point source emitters.

1.2.2. Total internal reflection

Under normal-incidence epi-illumination, the bulk of the excitation light passes through the sample, illuminating and exciting fluorophores throughout the thickness of the sample. This creates a fluorescent background from the out-of-focus excited fluorophores which reduces the signal-to-noise ratio (SNR) of the resulting image. Total internal reflection (TIR) is a technique that reduces the volume of the sample which is illuminated and improves the SNR of the image. It makes use of the difference between the refractive indices of the aqueous sample buffer and the glass of the coverslip.

When light is incident at the boundary between two materials of differing refractive index, it undergoes transmission across the interface and reflection back from it in proportions dependent on the angle of incidence and the values of the refractive indices. The

transmitted portion of the light is refracted at an angle θ_t from the normal according to Snell's Law:

$$n_i \sin \theta_i = n_t \sin \theta_t$$

Equation 1.2. Snell's Law

where n_i and n_t are the refractive indices of the incident and transmitting media respectively, and θ_i is the angle of incidence (see Figure 1.11)[50].

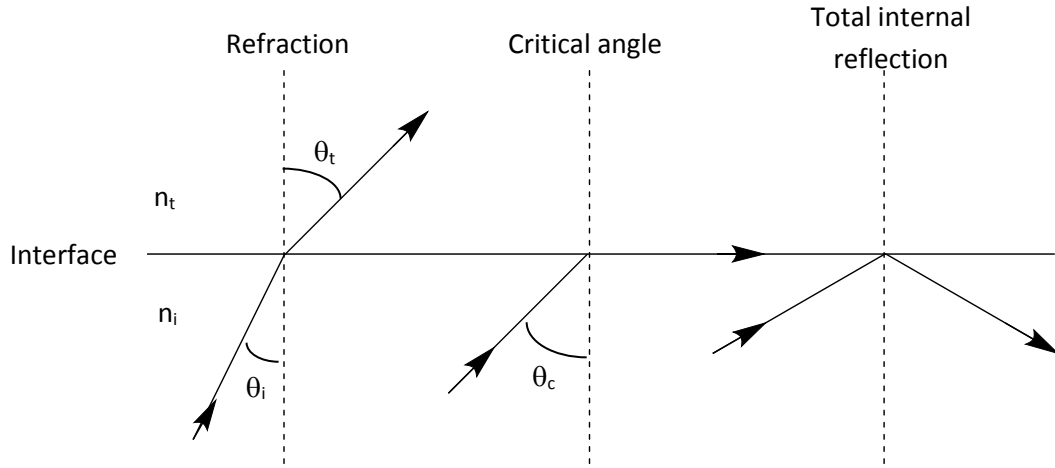


Figure 1.11. Schematics of refraction and total internal reflection

When $n_i > n_t$ (as for a glass-water interface: $n_{\text{glass}} = 1.52$, $n_{\text{water}} = 1.33$), θ_t increases as θ_i increases, as does the proportion of energy in the reflected beam, until θ_i reaches a point known as the critical angle, θ_c . At this point $\theta_t = 90^\circ$ and all of the incident light is reflected. Snell's Law can be rearranged to give an expression for the critical angle in terms of the relative index of refraction of the two media:

$$\theta_c = \sin^{-1} \frac{n_t}{n_i}$$

Equation 1.3. The critical angle for total internal reflection

Although all of the incident light energy is reflected at the critical angle, in order to satisfy the boundary conditions of the Fresnel equations describing the propagation of electromagnetic waves, there must exist a transmitted wave which on average does not carry energy across the boundary. This is known as the evanescent wave which propagates parallel to the surface in the plane of incidence. The intensity of the evanescent wave, $I(z)$, decays exponentially with perpendicular distance z from the interface[51]:

$$I(z) = I(0)e^{-\frac{z}{d}} \quad \text{where} \quad d = \frac{\lambda_0}{4\pi \sqrt{n_i^2 \sin^2 \theta_i - n_t^2}}$$

Equation 1.4. The relationship between the intensity of an evanescent wave and perpendicular distance from an interface

d is the penetration depth at which the intensity has decayed to $I(0)/e$.

λ_0 is the wavelength of the light in vacuum. Typically, the evanescent wave penetrates around 100 nm into the transmitting medium. This distance can be decreased by increasing the angle of incidence beyond the critical angle[51,52].

Various optical arrangements have been employed to achieve TIR in fluorescence microscopy. Broadly, they fall into two categories: prism-type TIR and objective-type TIR.

1.2.2.1. Prism-type total internal reflection

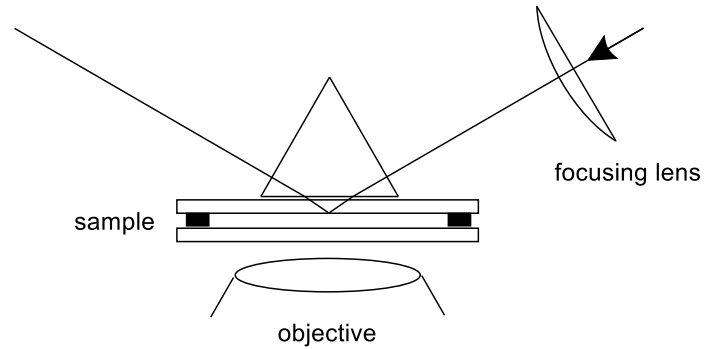


Figure 1.12. Schematic showing one possible optical arrangement for prism-type total internal reflection

Prism-type total internal reflection uses a prism to alter the angle at which incident light meets the coverslip-buffer interface. It does tend to restrict access to the sample and choice of objectives in some cases, however it produces a cleaner fluorescence than prism-less TIR and can be set up on an inverted or an upright microscope. In the example shown in Figure 1.12, the sample (consisting of two coverslips separated by a spacer ring) is placed face-down onto an inverted microscope objective with a prism placed on top. The prism is optically coupled to the upper coverslip with a layer of immersion oil or glycerol and excitation illumination is directed into the prism[52].

1.2.2.2. Objective-type total internal reflection

Objective-type total internal reflection uses a high NA objective to deliver light at a supercritical angle to the sample. By translating the excitation beam off-centre from the objective's optical axis, it is made to pass through the edge of the objective's pupil which increases the angle of incidence at the sample. This is illustrated in Figure 1.13.

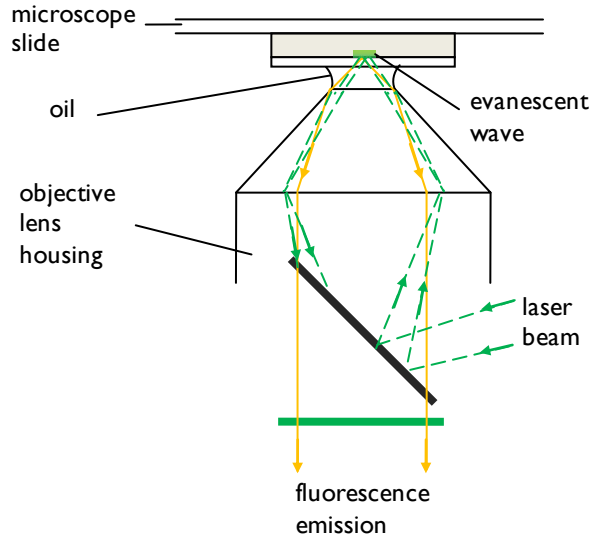


Figure 1.13. Schematic showing the optical arrangement for objective-type total internal reflection.

Increasing the radial off-set of the beam increases the angle of incidence. The largest angle (θ_m , measured from the optical axis) at which the beam can emerge into the immersion oil (refractive index n_{oil}) is determined by the numerical aperture, NA, of the objective:

$$NA = n_{oil} \sin \theta_m$$

Equation 1.5. Relationship between objective numerical aperture and the largest possible angle for an emergent beam

θ_m is equal to the angle of incidence at the coverslip-buffer interface, and n_{oil} is matched to n_i , the refractive index of the glass coverslip. By comparing Equation 1.3 and Equation 1.5 it can be seen that for TIR to occur ($\theta_m > \theta_c$), NA must be greater than n_t , the refractive index of the sample buffer. For typical high NA objectives (NA = 1.45) used with samples in aqueous buffers ($n \sim 1.33$), this poses no problem[52].

1.2.3. Image analysis techniques

1.2.3.1. Fluorophore localization

Super resolution and single molecule microscopy rely on the ability to locate the position of a particle to accuracies greater than the dimensions of its image. This is made possible using fitting and interpolation procedures for which the precision increases with the number of photons detected from the object of interest[53,54]. There are three main methodologies which have emerged for fluorophore localization: cross-correlation, centroid, and direct Gaussian fit[55]. Of these, the centroid and Gaussian fit methods work independently on each frame of an image series to provide direct sub-pixel localizations of a fluorophore while the cross-correlation method uses an ideal kernel image of the object of interest taken from one of the images in the series to guide the fitting procedure.

The centroid technique is computationally efficient and the simplest to implement, calculating the centre of a fluorescent image by weighting the pixels comprising it with

their respective intensity values and computing the average. This requires the use of thresholding to identify the pixels of the fluorescent image (having intensities greater than a specified value) from surrounding background. Binary thresholding results in a binary image where background pixel values are set to zero and image pixel values are set to one, in which case the centroid is calculated as the unweighted centre of the fluorescent image. Simple thresholding sets all pixel values found to be below the threshold to zero and retains the intensity values of the pixels found to be above the threshold. Cheezum et al.[55] noted that binary thresholding produced poorer fits to simulated particle positions than simple thresholding. Generally, binary thresholding is used as a first step to create a mask for particle identification which then enables weighted centroid calculation for each discrete region in the image. This technique has been used to track the Brownian motion of gold-labelled lipid molecules in membranes[56], and the movement of individual lipoprotein receptors within cell membranes as they bind fluorescently-labelled substrate[57].

The cross-correlation technique is the most computationally intensive of the three[58], requiring comparisons to be made between the current image and a kernel which contains the point spread function (PSF) of the object of interest, taken from one of the images in the series. The kernel is translated over the current image in single pixel increments and a correlation matrix is calculated for each position of the kernel until a maximum correlated image is found, corresponding to the maximum overlap of the kernel with the current particle image. The centre of the correlation image is found by the weighted centroid calculation. One of the early examples of this technique was tracking the movement of 150 nm-diameter beads attached to kinesin translating along a microtubule[58]. It ensures continuity between successive localizations by using a template image of the object of interest, but at the same time this makes it more susceptible to error if the object image changes shape or orientation between frames. This is not a problem for tracking bead movement viewed using differential interference contrast microscopy[58,59], but makes the technique less reliable in the case of point-like fluorescent dyes and proteins in lower SNR regimes subject to intensity fluctuations.

Direct Gaussian fitting is more sophisticated than centroid calculation. It approximates the shape of the PSF of a fluorophore, which for a point source viewed through a circular aperture is an Airy function, to a Gaussian function. Gaussian functions are much simpler and less computationally expensive to use than explicit expressions for PSFs. It has been shown that Gaussian approximations to PSFs achieve the same theoretical maximum localization accuracy as using more complete models of the PSF in the case of emitters with freely rotating dipoles[60]. This is true for wide field fluorescence microscopy in two dimensions[61]. Thresholding is often used for identification of fluorescent particles. The PSFs of single fluorophore images are then fitted to a two-dimensional Gaussian function to determine the intensity and position of the centre of the molecule. Thompson et al.[54] derived an expression for the standard error of the centre of the Gaussian distribution (σ_μ) which depends on the number of photons collected from the fluorophore (N) according to

$$\sigma_{\mu_i} = \sqrt{\frac{s_i^2}{N} + \frac{a^2/12}{N} + \frac{8\pi s_i^4 b^2}{a^2 N^2}}$$

Equation 1.6. Standard error of the mean for a Gaussian-approximated point spread function

where s_i is the standard deviation of the distribution in lateral direction i , a is the pixel size of the detector and b is the standard deviation of the background including background fluorescence noise and detector noise. Collecting a sufficient number of photons allows the position of a fluorophore to be calculated with arbitrarily high precision[53]. For the microscope system described in Section 3.3.1. typical values for a Cy3 dye imaged with an exposure time of 1 s are $N = 472$ photons, $s = 115$ nm, $a = 65$ nm, and $b = 3.6$ photons per pixel. Considering photon noise only (the first term in Equation 1.6), $\sigma_{\mu} = 5.3$ nm; the effect of pixelation (the second term) increases this value to 5.4 nm, while the background noise (final term) gives a final value of $\sigma_{\mu} = 9.4$ nm.

Of these three methods, cross-correlation and Gaussian fitting provide more accurate approximations to the sub-pixel position of a particle since they both use models of the PSF. Cross-correlation is susceptible to error in cases where the particle image changes shape or orientation between images. While direct Gaussian fitting is less reliable when viewing fluorescent molecules with fixed orientation dipoles, it is the most accurate representation of PSFs viewed under a wide field fluorescence microscope which remains computationally efficient to implement.

1.2.3.2. Step analysis

There are two main areas of single molecule research which deal with step counting in noisy data: monitoring the angular[62–64] or lateral[65–67] displacement of molecular machines, and determining the stoichiometry of protein complexes via photobleaching of fluorescent labels[43,68,69]. In the former case the steps represent displacement with time while in the latter they arise from discrete changes of fluorescence intensity with time.

Generally the displacement-time courses of molecular motors comprise many tens of steps which lends itself to the use of power spectra for identification of average step size. The occurrence of the unitary step size is first exaggerated by calculation of the pairwise distance distribution function (PDDF). This involves calculating the difference, ΔI_{ij} , for all data point pairs in the displacement/intensity trace $I(t)$ containing n data points as[70]:

$$\Delta I_{ij} = I(t_j) - I(t_i) \quad \text{where } i = 1:n, \quad j = 2:n, \quad j > i$$

Equation 1.7. Calculation of pairwise differences

These values are then arranged into a histogram which reveals a series of peaks spaced at intervals corresponding to multiples of the unitary step size. Taking the one-sided power spectrum of this histogram allows identification of the average unitary step in the frequency domain. This analysis method works well with data containing many equally-spaced steps. By converting the PDDF into the frequency domain, an automatic average is taken of all step sizes without the need to individually identify them. It does not extract information on dwell times at each distance/intensity level.

Another approach to step detection involves fitting steps to the trace and monitoring the progress of the fit[62,71,72]. This overcomes the problem of averaging step sizes encountered when using a PDDF with data exhibiting varying step sizes. Kerssemakers et al.[71] developed a step fitting algorithm which successively divides the trace into more steps, starting with a single step positioned at the point which gives the best least-squares fit to the data. This process continues while the quality of the fit is evaluated by comparing the current fit with a counter fit, which is a series of steps with positions in between those of the current best fit, and stops when the ratio between the Chi-squared values for the fit and counter fit is maximized. PIF, a step detection algorithm developed by McGuire et al.[72], works in the opposite direction by assigning many more steps than are present to start with and successively refining the fit according to an increasingly stringent threshold, using only the previous fit for each refinement.

Das et al.[73] employed a very simple method involving smoothing of the stepwise photobleaching trace using a simplified Chung Kennedy filter (a non-linear digital filter using forward and backward processing to avoid averaging over jump changes in a signal; this has been shown to reliably remove noise while preserving stepwise transitions[74–77]) followed by step identification which was defined as a drop in the intensity larger than two times the standard deviation of the preceding data points since the last step.

Some single molecule subunit counting studies have employed variations on the PDDF method to facilitate step identification in fluorescence intensity photobleaching traces[64,69,76]. Rather than perform a Fast Fourier Transform on the PDDF, the number of peaks is counted directly using a peak finding routine. This allows information on the step intensity levels to be extracted (being equal to the positions of the peaks), and reduces the possibility of mistaking transient increases in noise for step changes.

1.2.4. Determining protein quaternary structure

Single molecule subunit counting is a technique allowing the number of protein subunits within a complex to be determined directly. By labelling each subunit with a fluorescent molecule (either incorporating a fluorescent protein into the sequence of interest before expression or adding an organic dye later) the number of subunits can be deduced from the fluorescence emission of the labels even though the protein complex itself is smaller than the current spatial resolution of single molecule technologies[78]. Several different methods have been applied to the task of determining the number of fluorescently labelled subunits within a complex. These are stepwise photobleaching which counts the number of decreases in intensity as fluorescent labels photobleach[73,79], division of the starting intensity by the unitary step size[69,80], and ratiometric determination by comparison with a standard[81,82].

Both the stepwise photobleaching and intensity division methods rely on photobleaching the fluorescent labels to distinguish between them and determine the step size corresponding to a single fluorophore. Ratiometric determination uses a standard to determine the fluorescence intensity of a single label. The number of subunits in a complex is then calculated by dividing its recorded intensity by the standard value. Combining these methods with wide field fluorescence microscopy facilitates the study of many protein

complexes simultaneously. A distribution can be constructed to summarize the number of complexes observed containing a certain number of fluorescent labels (a label number distribution). It should be noted that the label number distribution is not interchangeable with a subunit number distribution due to experimental factors affecting the probability of detecting all fluorescent labels[78,83,84]. The label number distribution must be further analyzed to interpret the most likely structure of the underlying complex.

The main issue with all of the counting techniques is the possibility of missing events, which then skews the resulting label number distribution. The number of events can easily be miscalculated by the ratiometric method if the field of illumination is not even[85]. In the case of photobleaching, events might be missed if they occur close enough in time to appear as one large step. This problem becomes more prominent as the number of fluorescent labels within a complex increases. Ulbrich and Isacoff noted that for complexes containing more than 5 subunits the label number distributions began to look very similar for their experimental setup[79]. The maximum number of fluorescent labels observed directly by photobleaching to date is 12 Cy3 molecules labelling (probably two overlapping) packaging RNA rings[68].

Another source of missed events stems from the labelling method used. Labelling with fluorescent proteins ensures a 1:1 labelling efficiency where the fluorophore is encoded into the sequence for the protein of interest as a protein fusion tag[84]. However some fluorescent fusion proteins are not as reliable as organic dyes in terms of the proportion of the population which is actually fluorescent due to misfolding (the structure of a fluorescent protein is essential to its fluorescence; misfolding disrupts the arrangement of the residues responsible for fluorescence, and also the tertiary and/or quaternary structure which provide the necessary environment). The correct folding of fluorescent fusion proteins is dependent upon the temperature at which cells are cultured for expression[87] and the folding state of the fusion partner[88,89]. Equally, the solubility of the protein under study can be affected by the presence of a fluorescent fusion protein. Some fluorescent proteins are less sensitive to these factors, such as EGFP (enhanced GFP) which has improved maturation efficiency at 37 °C[90], or sfGFP (superfolder GFP) which folds well even when joined to a poorly-folded fusion partner[88]. When fusion proteins are misfolded, the population of viable subunits for counting experiments is reduced but the numbers of subunits observed in a complex are unaffected provided the fluorescent protein does not affect the natural function of the protein of interest. A problem with using a very stably folded fluorescent protein for subunit counting experiments is the potential increase in background fluorescence from misfolded subunits attached to functioning fluorescent proteins. While organic dyes are more reliable in terms of fluorescence, they do not guarantee a 1:1 labelling efficiency since they are added to the protein of interest post-expression. That is to say, the labelling efficiency is often only determined through bulk methods such as calculating molar ratios of protein to dye[91]. The average labelling efficiency achieved could be 1:1 but there is always the possibility of some unspecific labelling no matter how carefully the reaction conditions are controlled.

The possibility of missing photobleaching events causes the resulting label number distribution to follow a binomial distribution governed by the probability of detecting each

fluorescent label (θ)[73,79,84]. So long as experimental conditions are optimized to maximize this probability ($\theta \geq 0.7$), it is possible to unambiguously determine the number of subunits comprising the complexes under study from a binomial fit to the distribution for homogeneous complex populations[83].

1.2.5. Nucleotide binding and dynamics studies

Single molecule studies provide another means to probe the interaction of ligands with proteins. In contrast to bulk solution techniques such as equilibrium binding, fluorescence correlation spectroscopy and stopped flow measurements, single molecule techniques access binding dynamics between individual protein molecules and their ligands. This enables heterogeneities within a population to be seen which would be averaged out by bulk techniques[92,93].

Single molecule binding studies require the use of a fluorescent substrate[92,94,95]. The protein of interest is immobilized on a coverslip in a solution containing the fluorescent ligand. If the unbound ligand is fluorescent, it will produce a substantial background when illuminated under a fluorescence microscope. This necessitates the use of TIR illumination, creating an evanescent illumination field which only penetrates ~ 100 nm into the buffer from the coverslip[51,92]. So long as the diffusion rate of the fluorescent ligand is high enough compared to the exposure time used for image acquisition, freely diffusing fluorescent molecules will not be visible individually[95]. Binding events are then easily distinguished against the background as the occurrence of distinct fluorescent spots.

By measuring the times for which a ligand remains bound to a protein molecule, the off-rate constant for ligand binding can be accessed directly. This is calculated by creating a histogram of bound times for each protein molecule (or combining the bound times for several molecules if experimental conditions limit the number of binding events observed per molecule[92]) and fitting an exponential decay to the distribution. The off-rate binding constant is the inverse of the exponential time constant[96].

Analysis of the unbound times is more complicated owing to the fact that they might reflect a mixture of unbound times from two or more binding sites present in the same resolution-limited area[92,94]. In one study, the approximate halving of the unbound rate constant for certain fluorescent spots was taken as an indication that two protein molecules were present, and used as a rejection criterion[92]. This does not mean that the corresponding bound time rate constant is meaningless so long as concentrations of ligand are controlled to maximize the probability that no two binding events will occur at the same time for binding sites within the same resolution-limited area[94].

1.3. Aims

This thesis documents the use of single molecule techniques to address the unresolved questions of the stoichiometry of the ID complex and the kinetics of ATP binding to the AAA⁺ subunit ChII. This was accomplished by:

Adjusting an optical set-up to enable objective-type total internal reflection microscopy, required to reduce background fluorescence and improve the signal-to-noise ratio when observing single fluorescent molecules

Developing code in MATLAB to analyse image series, enabling the extraction of fluorophore intensities, the counting of stepwise decreases in intensity for subunit counting, and half amplitude threshold analysis to extract dwell times from nucleotide binding traces

Labelling subunits with organic dyes, investigating the resulting labelling efficiencies, checking chelatase activity of labelled subunits and determining optimum conditions for complex immobilization on a coverslip in order to perform single molecule subunit counting

Modelling stepwise photobleaching traces for comparison to experimental data, to further investigate results obtained from binomial fits to the observed fluorescent label number distributions

Proving interaction between fluorescent nucleotide and the nucleotide binding site of ChII and directly monitoring binding at a single molecule level in order to determine the off-rate constant

2. Materials and methods for biochemical analyses

2.1. Materials

2.1.1. Chemicals and reagents

All chemicals and reagents were purchased from Sigma-Aldrich Company Ltd. (Dorset, UK) and were of the highest life science or analytical grade unless stated otherwise. Alexa Fluor 532 and TNP-ATP were sourced from Fisher Scientific UK Ltd (Loughborough, UK). Cy3-EDA-ATP was purchased from Stratech Scientific (Newmarket, UK). Cy3 and Cy5 maleimide fluorescent dyes, and chromatographic columns and media for protein purification were obtained from GE Healthcare Life Sciences (Little Chalfont, UK), and porphyrins were purchased from Inochem, Ltd. (Carnforth, UK). Milli-Q® Integral ultrapure water (Milli-Q water) was produced to a purity of 18.2 MΩ·cm at 25 °C by further purifying distilled water using a Milli-Q® Integral Water Purification System (Millipore (UK) Limited, Watford, UK) and was the standard water source unless stated otherwise.

2.1.2. *E. coli* strains and plasmids

Escherichia coli Rosetta™(DE3)pLysS competent cells were originally obtained from Novagen (Millipore (UK) Limited, Watford, UK). Derivatives of Rosetta(DE3)pLysS and DH5α competent cells (a kind gift from J. Grasby, Department of Chemistry, University of Sheffield) were used in this work. These were propagated in preparation for chemical transformation using the Inoue method as detailed by Green and Sambrook (Chapter 3, Protocol 2)[97]. Stocks of chemically competent cells were stored at -80 °C.

The plasmids used in this work were prepared originally by Jensen et al.[9,11] These are pET9a-His₆ChID, pET9a-ChII, pET9a-His₆ChII and pET9a-His₆ChIH with the *Synechocystis* sp. PCC6803 chID, chII, chII, or chIH gene respectively cloned between the NdeI and BamHI sites. Plasmid DNA stocks were stored at -20 °C.

2.2. Preparation of growth media, reagents and buffers

Growth media and buffer chemicals were obtained from Fisher Scientific UK Ltd (Loughborough, UK). All work with cells was performed aseptically. Growth media, reagents and buffers were prepared as described by Green and Sambrook[97] unless stated otherwise.

Growth Media

Growth media were made up in deionised water. All containers, non-thermolabile growth media components and pipette tips used with *Escherichia coli* and DNA were sterilized by autoclaving at 15 p.s.i., 121 °C, for 15 minutes. Heat sensitive components were sterilized by passing through a 0.22 µm filter and added aseptically when the media had cooled to a temperature below 50 °C.

IPTG: Isopropyl-β-D-thiogalactopyranoside (IPTG) was dissolved in water to a concentration of 400 mM. The solution was filter sterilized, stored at 4 °C and used within 1 day.

LB medium: Luria-Bertani (LB) medium was prepared according to the protocol of Green and Sambrook (Appendix 1: Reagents and Buffers)[97] but omitting the pH adjustment step.

Super LB: Super LB medium was prepared in the same manner as LB Medium, but contained 20 g of tryptone, 10 g of yeast extract and 10 g of sodium chloride in 1 l of solution.

ZYM-5052 auto-inducing medium: ZYM-5052 auto-inducing medium was prepared as described by Studier[98].

Antibiotics: Chloramphenicol stock solutions were prepared as detailed by Green and Sambrook (Appendix 1: Reagents and Buffers)[97]. Neomycin stock solutions were prepared by dissolving 30 mg·ml⁻¹ of neomycin in water and filter sterilizing the solution. Stock solutions were stored at -20 °C.

Reagents

ATP and ADP: Adenosine 5'-triphosphate disodium salt (ATP) or adenosine 5'-diphosphate disodium salt (ADP) was dissolved in water to a concentration of approximately 60 mM. The pH was adjusted to between 7 and 8 with potassium hydroxide solution (1 M in water) before storage in small aliquots at -20 °C. The concentrations of the solutions were determined spectrophotometrically using the extinction coefficient given in Table 2.2.

D_{IX} and MgD_{IX}: Deuteroporphyrin IX dihydrochloride (D_{IX}) and magnesium(II) deuteroporphyrin IX dipotassium salt (MgD_{IX}) were each dissolved in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 34 °C) to an approximate concentration of 1 mM, vortexed for 1 min and 1 s respectively, and then centrifuged for 1 min at 12 100 g to remove any insoluble aggregates. Both solutions were protected from light, kept on ice, and used on the day they were made. The concentrations of the solutions were determined spectrophotometrically using the extinction coefficient for the Soret band of the porphyrin dications in 0.1 M aqueous hydrochloric acid (see Table 2.2).

DTT: Dithiothreitol (DTT) was dissolved in water to a concentration of 1 M. The solution was stored in 0.5 ml and 1 ml aliquots at -20 °C.

Gloxy: A 100× stock solution of glucose oxidase and catalase (Gloxy) was prepared as described by Selvin et al. (Protocol 4: Imaging Immobilized Cy3-DNA under Deoxygenated Conditions)[99], except that the buffer used was thiol-free chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH 7.7 at 25 °C), and the supernatant filtration step was omitted. Gloxy stock solution was flushed with a stream of nitrogen, protected from light, and stored at 4 °C for up to 3 months.

Glucose: D-glucose was dissolved in water at a concentration of 0.1 g·ml⁻¹. The solution was sterilized by passing through a 0.22 µm filter and stored at 4 °C.

MgCl₂·6H₂O: Magnesium chloride hexahydrate was dissolved in water to an approximate concentration of 1 M. The exact concentration of the solution was determined by complexometric titration with ethylenediaminetetraacetic acid (EDTA), using Eriochrome

Black T as an indicator. The solution was then diluted to a stock concentration of 83.5 mM, sterilized by passing through a 0.22 μ m filter, and stored at room temperature.

NADH: Nicotinamide adenine dinucleotide, (reduced form), disodium salt (Fisher Scientific UK Ltd, Loughborough, UK) was dissolved in water to a concentration of approximately 20 mM. Its exact concentration was determined spectrophotometrically using the extinction coefficient given in Table 2.2. The solution was stored in small aliquots at -20 °C.

PEP: Phospho(enol)pyruvic acid trisodium salt hydrate was weighed carefully and dissolved in water to a concentration of 200 mM. PEP solutions were stored in small aliquots at -20 °C.

Buffers

Buffers used in protein purification, kinetic work and single molecule microscopy were sterilized by passing through a 0.22 μ m filter and stored at 4 °C for up to a month.

2.3. Transformation of chemically competent *E. coli*

All transformed Rosetta(DE3)pLysS cells were grown in media containing 30 μ g·ml⁻¹ neomycin and 34 μ g·ml⁻¹ chloramphenicol for expression of magnesium chelatase subunits. Rosetta cells had previously been found to produce higher yields of the ChlD subunit than BL21(DE3), and were used for expression of all subunits to maintain continuity with prevailing lab protocols. All transformed DH5 α cells were grown in media containing 30 μ g·ml⁻¹ neomycin.

Transformation of competent cells (cells which are capable of taking up extracellular DNA from the surrounding environment) was performed in a manner similar to that outlined by Green and Sambrook (Chapter 3, Protocol 2)[97]. 50 μ l aliquots of competent cells in 1.5 ml ependorf tubes (suspended in Inoue transformation buffer: 10 mM PIPES/KOH, 250 mM KCl, 55 mM MnCl₂·4H₂O, 15 mM CaCl₂·2H₂O, pH 6.7 at 4 °C) were defrosted and left to rest on ice for 10 min. Approximately 10 ng of plasmid DNA was added to each tube of cells, which were then mixed gently by flicking and left to rest on ice for a further 30 min. The cells were subjected to heat shock at 42 °C for 90 s to induce DNA uptake and then cooled on ice for 2 min. 600 μ l of LB medium was added to each tube which was then incubated for 45 minutes at 37 °C with gentle agitation at 180 rpm for recovery. The cells were pelleted by centrifugation at 3600 g for 10 min and excess medium was poured away. The cell pellet was gently resuspended in the remaining LB medium (approximately 100 μ l) by pipetting. The cell suspension was subsequently transferred to and spread over agar LB medium containing the appropriate antibiotic. Agar plates were incubated overnight at 37 °C.

Cells transformed for protein expression were stored as glycerol stocks. Colonies from the agar plates were each transferred into 10 ml of LB medium containing appropriate antibiotic and grown overnight at 37 °C with shaking at 200 rpm. 800 μ l from the overnight growth was added to 200 μ l of sterile glycerol and mixed by vortexing. Glycerol stocks were stored at -80 °C.

2.4. Propagation of plasmid DNA

A culture of Rosetta(DE3)pLysS cells transformed with the required plasmid DNA was grown overnight at 37 °C with shaking at 200 rpm in 10 ml of LB medium containing 30 $\mu\text{g}\cdot\text{ml}^{-1}$ neomycin and 34 $\mu\text{g}\cdot\text{ml}^{-1}$ chloramphenicol. Cells were harvested from 5 ml of the culture by centrifugation at 12 000 g for 10 min at 4 °C. Plasmid DNA was isolated from the harvested cells using a QIAprep Spin Miniprep Kit (QIAGEN Ltd, Manchester, UK) and following the manufacturer's protocol. The success of the purification was verified by transforming the plasmid DNA into DH5 α cells (see Section 2.3.) and checking for colony growth after incubation overnight at 37 °C on agar LB medium containing 30 $\mu\text{g}\cdot\text{ml}^{-1}$ neomycin. Purified plasmid DNA was stored at -20 °C.

2.5. Expression of magnesium chelatase subunits

Rosetta(DE3)pLysS cells were used for protein expression. All media used for growing transformed Rosetta(DE3)pLysS cells contained 30 $\mu\text{g}\cdot\text{ml}^{-1}$ neomycin and 34 $\mu\text{g}\cdot\text{ml}^{-1}$ chloramphenicol.

2.5.1. ChII, His₆-ChII and ChIH

Cells transformed with the required plasmid DNA were streaked from glycerol stocks onto agar LB medium and grown overnight at 37 °C. Colonies from the plates were inoculated into 10 ml starter cultures of LB medium and grown overnight at 37 °C with shaking at 200 rpm. The starter cultures were added to 2 L baffled flasks containing 500 ml of ZYM-5052 auto-inducing medium and grown at 25 °C with agitation at 200 rpm for 21 h. Cells were harvested by centrifugation at 14000 g and 4 °C for 25 minutes, and then stored at -80 °C. 1 L of cell culture produced around 10 mg of ChII protein or 20 mg of ChIH protein after purification.

2.5.2. ChID

Starter cultures were grown following the same procedure as for expression of ChII and ChIH, then added to 2 L non-baffled flasks containing 500 ml of Super LB medium and incubated at 37 °C until an OD₆₀₀ of 0.8 to 1.0 was reached (after 3 to 5 h). ChID expression was induced by the addition of IPTG to a final concentration of 0.4 mM. The cells were grown for a further 16 h at 18 °C with shaking at 180 rpm, then harvested by centrifugation at 14000 g, 4 °C for 25 minutes and stored at -80 °C. 2 L of cell culture produced around 10 mg of ChID protein after purification.

2.6. SDS-polyacrylamide gel electrophoresis

Proteins were separated using sodium dodecyl sulphate polyacrylamide gel electrophoresis (SDS-PAGE) mini-gels with a 5 % stacking gel. A 10 % separating gel was used for ChII and ChID, and an 8 % separating gel was used with ChIH unless stated otherwise. 0.75 mm thick mini-gels were prepared and run using the Mini-PROTEAN 3 system (Bio-Rad Laboratories Ltd., Hemel Hempstead, UK) and following the protocol described by Kielkopf et al. (Protocol 8: SDS-PAGE of Proteins)[100] with minor alterations. If protein was to be visualized via fluorescence as well as or instead of staining, 2,2,2-trichloroethanol (TCE) was added to the separating gel at 0.5 % of the separating gel volume to create a stain-free gel as described by Ladner et al.[101]

Protein samples were made up in SDS gel-loading buffer containing β -mercaptoethanol and EDTA (50 mM Tris-HCl (pH 6.8), 1.6 % (w/v) SDS, 8 % (v/v) glycerol, 3 % (v/v) β -mercaptoethanol, 12.5 mM EDTA, 0.02 % w/v bromophenol blue) and heated at 75 °C for 10 min before loading. In the case of proteins conjugated with fluorescent dyes, samples were incubated with the SDS gel-loading buffer for 10 min at room temperature to limit protein-dye dissociation. Precision Plus Protein™ Unstained Standards (10 kDa to 250 kDa; Bio-Rad Laboratories Ltd.) or the Mark12™ Unstained Standard (2.5 kDa to 200 kDa; Life Technologies Ltd, Paisley, UK) were used as molecular weight markers. Tris Glycine SDS PAGE Buffer (10×) was purchased from Scientific Laboratory Supplies Limited (Nottingham, UK). Gels were run for 45 min at a voltage of 200 V.

When proteins were fluorescently labelled, the fluorescent dye was imaged under UV illumination before staining. Protein bands were then visualized with colloidal Coomassie G-250 (SimplyBlue™ SafeStain: Life Technologies Ltd.) according to the manufacturer's instructions. Protein in stain-free gels was visualized by irradiation with UV illumination for 5 min, during which time the protein bands begin to fluoresce. After fluorescence visualization, stain-free gels could be stained as normal. Gel visualization was performed on a ChemiDoc™ MP imaging system (Bio-Rad Laboratories Ltd.).

2.7. Purification of magnesium chelatase subunits

2.7.1. His₆-tagged proteins

His₆-tagged proteins were purified essentially as described by Jensen et al.[17] They underwent an initial purification step via immobilized metal affinity chromatography (IMAC). Pellets from 1 l of cell culture (or 2 l of cell culture in the case of ChlD) were defrosted and resuspended in 20 ml of chilled imidazole binding buffer (25 mM Tris/NaOH, 0.5 M NaCl, 0.3 M glycerol, 5 mM imidazole, 1 mM AEBSF, pH 7.4 at 4 °C). Cells were lysed by sonication on ice in 4 bursts of 15 s each separated by rest periods of 30 s using a Soniprep 150 (MSE (UK) Ltd, London, UK). Cell debris was pelleted by centrifugation at 40 000 g for 40 min at 4°C. The supernatant was passed through a 0.45 μ m filter before being loaded onto a 5 ml HisTrap HP column pre-equilibrated with 6 column volumes of imidazole binding buffer. The column was washed with 10 column volumes of imidazole wash buffer (25 mM Tris/NaOH, 0.5 M NaCl, 0.3 M glycerol, 20 mM imidazole, pH 7.4 at 4 °C) before the protein was eluted with imidazole elution buffer (25 mM Tris/NaOH, 0.1 M NaCl, 0.3 M glycerol, 400 mM imidazole, pH 7.4 at 4 °C) in 1.5 ml fractions. Eluted protein was detected using Bradford Reagent and relative molecular mass and purity were confirmed via SDS-PAGE.

Protein fractions eluted from the HisTrap HP column were combined, diluted to a final volume of 50 ml with Tricine buffer A (50 mM Tricine/NaOH, 0.3 M glycerol, 1 mM DTT, pH 7.9 at 4 °C), and further purified to remove any ATPase contaminants by high resolution anionic exchange chromatography. The sample was loaded onto a 2.5 ml Tricorn 5/150 column packed with Source 15Q and pre-equilibrated with 5 column volumes of Tricine buffer A using an ÄKTA pure FPLC system (GE Healthcare Life Sciences). The sample was washed with 5 column volumes of the same buffer before the proportion of Tricine buffer B (50 mM Tricine/NaOH, 0.3 M glycerol, 1 M NaCl, 1 mM DTT, pH 7.9 at 4 °C) was raised to 20

% with a linear gradient over 1 column volume. The column was then washed with 5 volumes of 20 % buffer B. Protein was eluted in 2 ml fractions using another linear gradient increasing from 20 % to 100 % Tricine buffer B over 15 column volumes. All subunits eluted at approximately 0.3 M NaCl. Eluting protein was detected via UV absorption at 280 nm and relative molecular mass and purity were confirmed via SDS-PAGE. Unless the subunit being purified was ChII, protein fractions were then stored in 0.2 ml to 1 ml aliquots at -80 °C.

In the case of ChII, the protein was subjected to a final cleaning step by gel filtration as described by Reid et al.[12] The fractions eluted from the Source 15Q column were combined and spin concentrated to around 2 ml. The sample was loaded onto a 320 ml HiLoad™ 26/600 Superdex™ 200 pg column pre-equilibrated with 1 volume of Tricine buffer A and eluted at 0.5 ml·min⁻¹ in 10 ml fractions. Protein eluted at around 175 ml and was detected by UV absorbance at 280 nm and confirmed via SDS-PAGE. The fractions were stored in 0.2 ml to 1 ml aliquots at -80 °C.

2.7.2. Non-tagged proteins

Non-tagged proteins were purified as described previously[12], with the addition of an initial step using low resolution anionic exchange chromatography. Pellets from 1 l of cell culture were defrosted and resuspended in 20 ml of chilled Tricine binding buffer (50 mM Tricine/NaOH, 0.3 M glycerol, 1 mM AEBSF, 1 mM DTT, pH 7.9 at 4 °C). The cells were lysed by sonication and the protein extracted in the supernatant as described in Section 2.7.1. The supernatant was then passed through a 0.45 µm filter and loaded onto a 10 ml HiTrap Q FF column pre-equilibrated with 5 column volumes of Tricine buffer A (50 mM Tricine/NaOH, 0.3 M glycerol, 1 mM DTT, pH 7.9 at 4 °C) using an ÄKTA pure FPLC system. After washing with a further 5 column volumes of Tricine buffer A, the proportion of Tricine buffer B (50 mM Tricine/NaOH, 0.3 M glycerol, 1 M NaCl, 1 mM DTT, pH7.9 at 4 °C) was increased to 20 % using a linear gradient over 1 column volume. The column was then washed with 4 volumes of 20 % Tricine buffer B before the protein was eluted in 2 ml fractions using another linear gradient increasing from 20 % to 100 % Tricine buffer B over 15 column volumes. ChII subunits eluted at around 0.3 M NaCl, detected by UV absorbance at 280 nm. Relative molecular mass and purity were confirmed via SDS-PAGE.

Protein fractions from the HiTrap Q FF column were combined and diluted to a final volume of 50 ml with Tricine buffer A before further purification via high resolution anionic exchange and gel filtration chromatography as described previously in Section 2.7.1. The fraction showing highest chelatase activity was retained and stored in 0.2 ml to 1 ml aliquots at -80 °C.

Defrosted protein was centrifuged at 12 100 g for 10 min at 4 °C, then transferred to new ependorf tubes to remove aggregates before use.

2.8. Protein, fluorescent dye and substrate concentration determination

2.8.1. UV-Vis absorbance

Concentrations of pure protein were determined spectrophotometrically via absorbance at 280 nm on a NanoDrop 2000 Spectrophotometer (Thermo Scientific NanoDrop products, Wilmington, US) using the Protein A_{280} application with a sample volume of 2 μ l. A blank was established from a scan of 2 μ l of the appropriate buffer over the range 220 nm to 350 nm. The sample absorbance spectrum was measured over the same range, corrected for the blank, and returned normalized to a 10 mm path length with baseline correction at 340 nm. This provided both A_{280} and A_{260} absorbance readings.

Concentrations of protein labelled with fluorescent dyes showing negligible absorption at 280 nm in the absence of nucleotide were determined via absorbance measurements using the Proteins & Labels application of the NanoDrop 2000 with a sample volume of 2 μ l. This application includes a correction for the contribution of the dye absorbance at 280 nm and comes with pre-defined settings for Cy3 and Cy5 dyes. The details of other dyes (extinction coefficient at the appropriate absorption wavelength and A_{280} correction factor) were entered manually and saved for subsequent use. A blank was established from a scan of 2 μ l of the appropriate buffer over the range 220 nm to 750 nm. The extinction coefficient and relative molecular mass of the protein were entered manually before a measurement. The sample absorbance spectrum was measured, corrected for the blank, and returned normalized to a 10 mm path length with baseline correction at either 340 nm or 750 nm depending on the absorbance spectrum of the dye. This provided concentration values for both the dye and the protein. The extinction coefficients and molecular masses of the proteins are given in Table 2.1.

Plasmid DNA concentrations were determined via absorbance at 260 nm using the Nucleic Acid application with a sample volume of 2 μ l.

Table 2.1. Molecular masses and extinction coefficients of magnesium chelatase subunits

Subunit	Molecular mass (kDa) ^a	ϵ_{280} ($M^{-1}\cdot cm^{-1}$) ^b
ChII	39.463	11290
ChID	73.665	44560
ChIH	148.555	163330

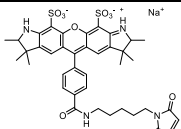
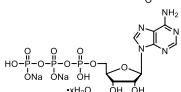
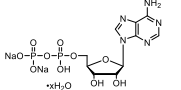
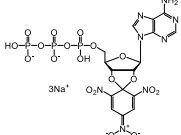
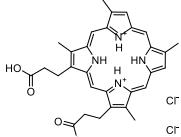
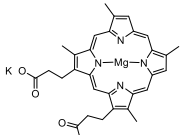
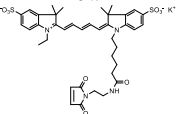
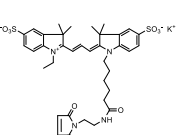
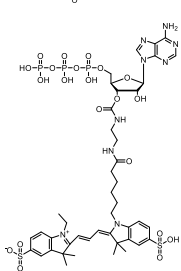
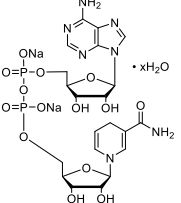
^aTaken from the corresponding UniProtKB database entry[102]

^bDetermined by Reid and Hunter[29]

Concentrations of fluorescent dyes and substrates were determined via absorbance measurements at various wavelengths using either a Cary 50 UV-Vis Spectrophotometer (Agilent Technologies LDA UK Limited, Stockport, UK) or the UV-Vis application on a NanoDrop 2000 Spectrophotometer (which provides the functionality of a conventional spectrophotometer for sample volumes of 2 μ l). The extinction coefficients and molecular masses of the dyes and substrates are given in Table 2.2.

Table 2.2. Molecular masses and extinction coefficients of fluorescent dyes and substrates.

The solvent listed is that in which the absorption measurements were performed. The absorption wavelength provided is the absorption maximum in the corresponding solvent.

Molecule		Molecular mass (kDa)	Solvent	Absorption wavelength	Extinction coefficient ($M^{-1}\cdot cm^{-1}$)
Alexa Fluor 532 C ₅ maleimide ^a		812.88	MeOH	528	78 000
ATP ^b		551.1	H ₂ O	259	15 400
ADP ^b		471.2	H ₂ O	259	15 400
TNP-ATP ^c		784.22	Aqueous solution pH 8	408	26 000
D ₁₉		583.51 ^e	0.1 M aqueous HCl ^d	398 ^d	433 000 ^d
MgD ₁₉		609.05 ^e	0.1 M aqueous HCl ^d	398 ^d	433 000 ^d
Cy5 maleimide ^e		817	Aqueous buffer pH 7.0 to pH 7.5	649	250 000
Cy3 maleimide ^e		791	Aqueous buffer pH 7.0 to pH 7.5	550	150 000
Cy3-EDA-ATP ^e		1206.03	Tris-HCl pH 7.5	550	150 000
NADH ^e		709.4	H ₂ O	340	6 300

^aTaken from Section 2.2 of The Molecular Probes® Handbook[103]

^bTaken from Engel[104]

^cTaken from Section 17.3 of The Molecular Probes® Handbook[103]

^dTaken from Falk (Appendix 1: Absorption Spectra)[105]

^eTaken from the product specification sheet

2.8.2. Bradford Assay

The concentration of proteins labelled in the presence of nucleotide with fluorescent dyes showing negligible absorbance at 595 nm (Fluorescein C₅ maleimide) was determined via the Bradford Assay[106] in a microwell plate format using a FLUOstar OPTIMA microplate reader (BMG LABTECH Ltd., Aylesbury, UK). 10 µl of bovine serum albumin (BSA) standards in the range 0 mg·ml⁻¹ to 1.4 mg·ml⁻¹ were added to 190 µl of Bradford Reagent to produce a calibration curve, according to the supplier's instructions.

2.9. Magnesium chelatase assay

The steady state rate of MgD_{IX} production by the magnesium chelatase subunits in the presence of the substrates MgATP²⁻, Mg²⁺, and D_{IX} was determined in a microplate reader assay.

2.9.1. Assay conditions

Solutions of ATP, MgCl₂, MgD_{IX}, and D_{IX} were prepared as described in Section 2.2. The concentration of ATP, MgD_{IX}, and D_{IX} solutions was determined spectrophotometrically as described in Section 2.8.1. Assays were carried out in 100 µl volumes containing chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 34 °C), 8 µM D_{IX}, 15 mM MgCl₂, and 5 mM ATP. MgCl₂ was kept at an excess of at least 1 mM over ATP to ensure that essentially all ATP was maintained in the MgATP²⁻ form. KCl was added to maintain the ionic strength at 0.1. The reaction mixtures were added to a microwell assay plate and pre-warmed at 34 °C for at least 5 minutes before the enzyme subunits were added to initiate the reaction, to final concentrations of 0.1 µM ChII, 0.1 µM ChID and 0.4 µM ChIH.

The concentration of the MgD_{IX} product was monitored at 34 °C over a period of 1 h or 2 h directly via its emission at 580 nm (emission filter (580 ± 5) nm) when excited at 420 nm (excitation filter (420 ± 5) nm) in a FLUOstar OPTIMA microplate reader (BMG LABTECH Ltd., Aylesbury, UK). Calibration curves were constructed in duplicate or triplicate from 100 µl volumes containing chelatase buffer, enzyme in the previously mentioned concentrations, and MgD_{IX} varying in concentration from 0 µM to 8 µM. The Optima MARS data analysis suite (version 2.20) provided with the platereader was used to perform a second degree polynomial fit to the mean calibration curve and convert the measured fluorescence emission values to MgD_{IX} concentrations. The steady state rate of MgD_{IX} production was calculated from the maximum gradient of the progress curve.

2.9.2. Data analysis

The steady state rate of MgD_{IX} production (v_{ss}) was plotted against MgD_{IX} concentration ([S]) and fitted using Igor Pro 6.2 (WaveMetrics, Inc., Lake Oswego, OR, US) to find the maximum rate at saturating substrate concentrations (V_{max}) and the Michaelis constant K_m according to the Michaelis-Menten equation:

$$v_{ss} = \frac{(V_{max}[S])}{(K_m + [S])}$$

Equation 2.1. The Michaelis-Menten equation.

2.10. ATPase assay

The steady-state rate of ATP hydrolysis by ChII was determined in a coupled assay. Solutions of ATP and MgCl₂ were prepared as described in Section 2.2. The concentration of the ATP solution was determined spectrophotometrically as described in Section 2.8.1.

Assays were carried out in 100 µl volumes containing chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 34 °C), 15 mM MgCl₂, and 5 mM ATP with 2 mM PEP, 200 µM NADH, 2 U of pyruvate kinase (PK) and 2 U of lactate dehydrogenase (LDH). MgCl₂ was kept at an excess of at least 1 mM over ATP to ensure that essentially all ATP was maintained in the MgATP²⁻ form. KCl was added to maintain the ionic strength at 0.1. The reaction mixtures (minus the ATP) were added to a microwell assay plate and pre-warmed at 34 °C for 10 minutes before ChII was added to a final concentration of 1 µM and ATP was added to initiate the reaction.

The progress of the assay was monitored at 34 °C over a period of 40 min or 60 min using a FLUOstar OPTIMA microplate reader to record the decrease in absorbance at 340 nm on oxidation of NADH. Calibration curves were constructed in triplicate from 100 µl volumes containing chelatase buffer and NADH varying in concentration from 0 µM to 225 µM. The Optima MARS data analysis suite (version 2.20) provided with the platereader was used to perform a linear regression fit to the mean calibration curve and convert the measured absorbance values to ATP concentrations. The steady state rate of ATP hydrolysis was determined from the gradient of the linear progress curves.

2.11. Fluorescent nucleotide titrations

The dissociation constant for TNP-ATP binding to ChII was determined from a titration assay exploiting the fluorescence increase observed in TNP-ATP upon binding to protein. The dissociation constant for 3'-O-Cy3-EDA-ATP binding to ChII was determined from a competitive titration assay where Cy3-ATP was used to displace TNP-ATP with a resultant fluorescence decrease.

2.11.1. TNP-ATP titration assay

The concentration of TNP-ATP was held constant at 2 µM while the concentration of ChII was varied between 0 µM and 30 µM. Each titration point was performed separately in a 100 µl volume of chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 °C) containing 10 mM free Mg²⁺. The ionic strength was adjusted to 0.1 with KCl. Samples were incubated for 5 min at 25 °C before the fluorescence emission spectrum of TNP-ATP was recorded on a FluoroMax 3 spectrofluorometer (Horiba UK Limited, Stanmore, UK) with excitation at 408 nm and detection from 515 nm to 575 nm.

The area under the fluorescence emission spectrum for each titration point was calculated and plotted against protein concentration. A single site binding model was fitted to the

data using Igor Pro 6.2 (WaveMetrics, Inc., Lake Oswego, OR, US) to find the dissociation constant, K_d :

$$\Delta F = \Delta F_{max} \frac{(L_t + E_t + K_d) - \sqrt{(L_t + E_t + K_d)^2 - 4L_tE_t}}{2L_t}$$

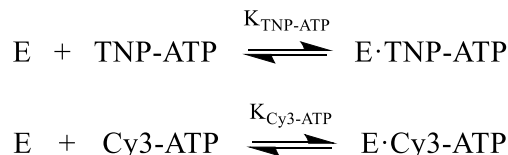
Equation 2.2. Single site binding model for spectroscopically distinct bound and unbound ligand

where ΔF is the change in fluorescence intensity relative to the zero protein sample, ΔF_{max} is the maximum change in fluorescence intensity (treated as a variable during fitting), L_t is the concentration of TNP-ATP, and E_t is the concentration of ChII.

2.11.2. Cy3-ATP competition titration

The competition titration was performed in a single cuvette with constant volume addition of Cy3-ATP. The starting volume of 100 μ l contained 5 μ M TNP-ATP and 5 μ M ChII in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 °C) with 10 mM free Mg^{2+} and ionic strength of 0.1. 10 μ l volumes of 10 μ M Cy3-ATP were added until the final concentration of TNP-ATP, ChII and Cy3-ATP was each 3.3 μ M. After each addition the sample was incubated for 5 min at 25 °C before the fluorescence emission from TNP-ATP was recorded on a FluoroMax 3 spectrofluorometer (Horiba UK Limited, Stanmore, UK) with excitation at 408 nm and detection from 515 nm to 575 nm.

The emission spectra were corrected for the contribution from Cy3-ATP and the fluorescence value for each titration point was calculated as the area under the corrected spectrum. Dynafit[107] was used to calculate the dissociation constant for Cy3-ATP according to the competitive binding mechanism



Scheme 2.1. Competitive binding mechanism

2.12. Pull downs

The dependence of ChII and ChID subunit interactions on the presence of nucleotide was investigated via pull down assays. Pull down assays were performed in batch format using Ni-NTA Agarose beads (Sepharose CL-6B support; QIAGEN Ltd., Manchester, UK) which bound His₆-ChID subunits. If non-tagged ChII bound to the immobilized His₆-ChID in the presence or absence of nucleotide, it would remain on the beads through successive washes and show up in SDS-PAGE gels, indicating an interaction between the subunits.

For each sample, 50 μ l of Ni-NTA bead slurry (1:1 mixture (v/v) of beads and 20 % ethanol) was exchanged into imidazole binding buffer (50 mM MOPS/NaOH, 0.1 M NaCl, 10 mM $MgCl_2$, 0.3 M glycerol, pH 7.7 at 25 °C): the slurry was centrifuged for 1 min at 3600 g to pellet the beads, the supernatant was removed, and 50 μ l of imidazole binding buffer was added and mixed by gentle vortexing. This procedure was repeated twice more for a total

of 3 washes. The beads were then pelleted again and changed into 25 μ l of imidazole binding buffer containing the appropriate nucleotide (none, 100 μ M TNP-ATP, or 5 mM ATP). The $MgCl_2$ concentration was 15 mM in samples containing 5 mM ATP to keep free Mg^{2+} at 10 mM.

A 200 μ l sample containing 5 μ M each of ChII and His₆-ChID with the appropriate nucleotide in imidazole binding buffer was added to the corresponding bead slurry and set on an end-over-end mixer at room temperature for 10 min. The beads were then washed 4 times with 200 μ l of imidazole binding buffer containing the appropriate nucleotide each time. Finally the beads were pelleted to remove the supernatant, mixed with 20 μ l of SDS-PAGE gel-loading buffer and heated for 10 min at 90 °C to elute the protein. The supernatant was run on an SDS-PAGE gel to determine the subunits present.

2.13. Fluorescent labelling of protein

Subunits were labelled with fluorescent dyes via maleimide linkage to cysteine residues following manufacturers' protocols. Generally, fluorescent dye (dissolved in dimethyl sulfoxide (DMSO) or dimethylformamide (DMF)) was added to protein in thiol modification buffer and left for 2 h at room temperature. Unreacted dye was quenched by the addition of DTT and separated from the labelled protein by gel permeation chromatography.

2.13.1. Fluorescent dye preparation

All fluorescent dyes and stock solutions prepared from them were protected from light by wrapping in tin foil. Work with fluorescent dyes was carried out under low light levels.

Fluorescein-5-Maleimide was dissolved in DMSO to a concentration of 20 mM, stored at -20 °C and used as required.

Cy3 and Cy5 dyes were prepared fresh for each labelling reaction in DMF.

2.13.2. Protected single labelling of ChII

ChII contains 4 cysteine residues, all but one of which are protected from reaction by maleimide linkage when in the presence of nucleotide. Defrosted protein was concentrated to between 50 μ M and 100 μ M and buffer exchanged into thiol modification buffer (50 mM Tricine/NaOH, 0.3 M glycerol, 150 mM NaCl, pH7.4 at 20 °C) containing 5 mM ADP and 15 mM $MgCl_2$. Fluorescent dye was added drop-wise to a molar excess of 10 to 20 against the protein concentration. If the dye was dissolved in an organic solvent, the volume added did not exceed 5 % of the protein solution volume to avoid precipitation of the protein. The reaction was left to proceed in the dark at room temperature with stirring for 2 h. In the case of Cy- dyes, after 2 hr at room temperature the reaction was then left to run overnight at 4 °C. Excess thiol-reactive reagent was consumed by addition of DTT to a final concentration of 10 mM.

2.13.3. Labelling ChID

ChID was labelled in the absence of nucleotide. The labelling procedure was the same as outlined in Section 2.13.2. except the thiol modification buffer used did not contain ADP or $MgCl_2$.

2.13.4. Desalting to remove unreacted dye

Labelled protein was separated from excess, unconjugated dye by gel permeation chromatography. The reaction mixture was passed through a 10 ml HiTrap Desalting column (Sephadex G-25 Superfine medium; GE Healthcare Life Sciences, Little Chalfont, UK) pre-equilibrated with 5 column volumes of thiol modification buffer containing 1 mM DTT. Eluted protein was collected in 1 ml fractions.

2.13.5. Checking conjugation and determining labelling efficiency

Successful conjugation was confirmed by SDS-PAGE as described for labelled proteins in Section 2.6. Protein and dye concentration was determined as described in Section 2.8. The average labelling efficiency was estimated from the molar concentration ratio of the protein and the dye:

$$\text{Labelling efficiency (dye molecules per protein molecule)} = \frac{\text{dye concentration } (\mu\text{M})}{\text{protein concentration } (\mu\text{M})}$$

Equation 2.3. Calculation of labelling efficiency

2.14. Purification of Cy3-EDA-ATP

The isomers of Cy3-EDA-ATP were separated via HPLC as described by Oiwa et al.[94] The nucleotide was eluted isocratically from a 3.9 mm × 150 mm Nova-Pak C18 column (Waters Limited, Elstree, UK) pre-equilibrated for 30 min with TEAB buffer (100 mM TEAB, 12 % acetonitrile, pH 7.4 at 25 °C) at 1.5 ml·min⁻¹. It was detected by UV absorption at 260 nm and eluted after about 20 min. Once separated, the isomers were rapidly concentrated using a rotary evaporator; TEAB buffer was co-evaporated with acetone. The resulting sample was freeze dried over 2 days, then dissolved in water and stored, protected from light, at -80 °C.

The purity of the isomers thus obtained was checked by eluting the samples as described before. The peaks in the HPLC absorbance records were fitted to exponentially modified Gaussian distributions using the Multipeak Fitting 2 package provided with Igor Pro (WaveMetrics, Inc., Lake Oswego, OR, US) to determine the purity of the isomers. A check was also made to ensure that the ATP had not become hydrolysed during the process of drying the samples. This was carried out as described by Oiwa et al.[94]: nucleotide was eluted isocratically from a Nova-Pak C18 column pre-equilibrated for 30 min with phosphate buffer (10 mM KH₂PO₄/K₂HPO₄, 11 % acetonitrile, pH 6.8 at 25 °C) at 0.5 ml·min⁻¹.

2.15. Single molecule studies

See Sections 3.2. and 3.3. for a description of the optical setup used and preparation of sample chambers.

2.15.1. Stoichiometry measurements

Sample solutions consisted of magnesium chelatase subunits subjected to rapid serial dilution in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH 7.7 at 25 °C) to a final concentration of 0.01 nM to 0.1 nM. When inter-subunit interactions were under investigation, the subunits were incubated for 5 minutes at room temperature at a concentration of at least 3 μM in the presence of 5 mM ATP and 15 mM MgCl₂ to allow

complex formation before rapid serial dilution in chelatase buffer containing ATP and MgCl₂ and immobilization on the cover slip. Imaging buffer (50 mM MOPS/KOH, 0.3 M glycerol, 5 mM ATP, 15 mM MgCl₂, 1× Gloxy, 0.8 % (w/v) D-glucose, pH 7.7 at 25 °C) was then exchanged into the sample chamber.

2.15.2. Nucleotide binding assays

Sample solutions consisted of 1 μM protein in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH 7.7 at 25 °C). Once the protein was immobilized on the surface the sample was exchanged into imaging buffer containing 30 nM to 100 nM fluorescent nucleotide (50 mM MOPS/KOH, 0.3 M glycerol, 10 mM MgCl₂, 1× Gloxy, 0.8 % (w/v) D-glucose, pH 7.7 at 25 °C).

3. Methods and experimental setup for single molecule microscopy

3.1. Equipment and materials

See Section 2.2. for details of buffer and solution preparations used in Section 3.2.3. Solvents and reagents used in the cleaning and functionalization of microscope coverslips were of the highest analytical grade and were obtained from Sigma Aldrich. Milli-Q® Integral ultrapure water (Milli-Q water) was produced to a purity of 18.2 MΩ·cm at 25 °C by further purifying distilled water using a Milli-Q® Integral Water Purification System (Millipore (UK) Limited, Watford, UK). Details of equipment used in the optical bench setup for single molecule microscopy experiments are provided in Table 3.1.

Table 3.1. Details of equipment used in the optical bench setup for single molecule microscopy experiments

Item	Supplier	Details
Consumables		
Microscope slide coverslips	Fisher Scientific Ltd, Loughborough, UK	No. 1.5 borosilicate glass
Microscope slides	Fisher Scientific Ltd, Loughborough, UK	Plain glass 1.0 to 1.2 mm thick
Equipment		
Inverted fluorescence microscope	Nikon UK Limited, Kingston Upon Thames, UK	Eclipse Ti-E inverted microscope system (with Ti-LAPP Modular Illumination System TIRF module for nucleotide binding work)
sCMOS camera	Hamamatsu Photonics UK Limited, Welwyn Garden City, UK	ORCA-Flash4.0 V2 Digital CMOS camera, 4.0 megapixels, 6.5 μm pixel size
sCMOS camera	Andor Technology Ltd., Belfast, UK	Zyla 4.2 sCMOS, 4.2 megapixels, 6.5 μm pixel size
Microscope 60x objective	Nikon UK Limited, Kingston Upon Thames, UK	CFI Apo TIRF 60× Oil, oil immersion, NA 1.49
Microscope 100x objective	Nikon UK Limited, Kingston Upon Thames, UK	CFI Apo SR TIRF 100× Oil, oil immersion, NA 1.49
Lasers (405 nm, 647 nm)	Coherent UK Ltd., Ely, UK	OBIS CW solid state laser system
Laser (532 nm)	Laser 2000 (UK) Ltd., Huntingdon, UK	100 mW solid state diode laser
Laser (514 nm)	Coherent UK Ltd., Ely, UK	OBIS CW solid state laser system
Dichroic mirrors (545/650 nm dual-edge, 532 nm single-edge)	Laser 2000 (UK) Ltd., Huntingdon, UK	Semrock 545/650 nm BrightLine® dual-edge dichroic beamsplitter, Semrock 532 nm laser BrightLine® single-edge laser-flat dichroic beamsplitter
Emission filters (565/24 nm bandpass, 488/647 nm dual-notch)	Laser 2000 (UK) Ltd., Huntingdon, UK	Semrock 565/24 nm BrightLine® single-band bandpass filter, 488/647 nm StopLine® dual-notch filter
Cy5 filter cube	Nikon UK Limited, UK	Cy5 HYQ (Bandpass Emission)

3.2. Preparation of microscope slides

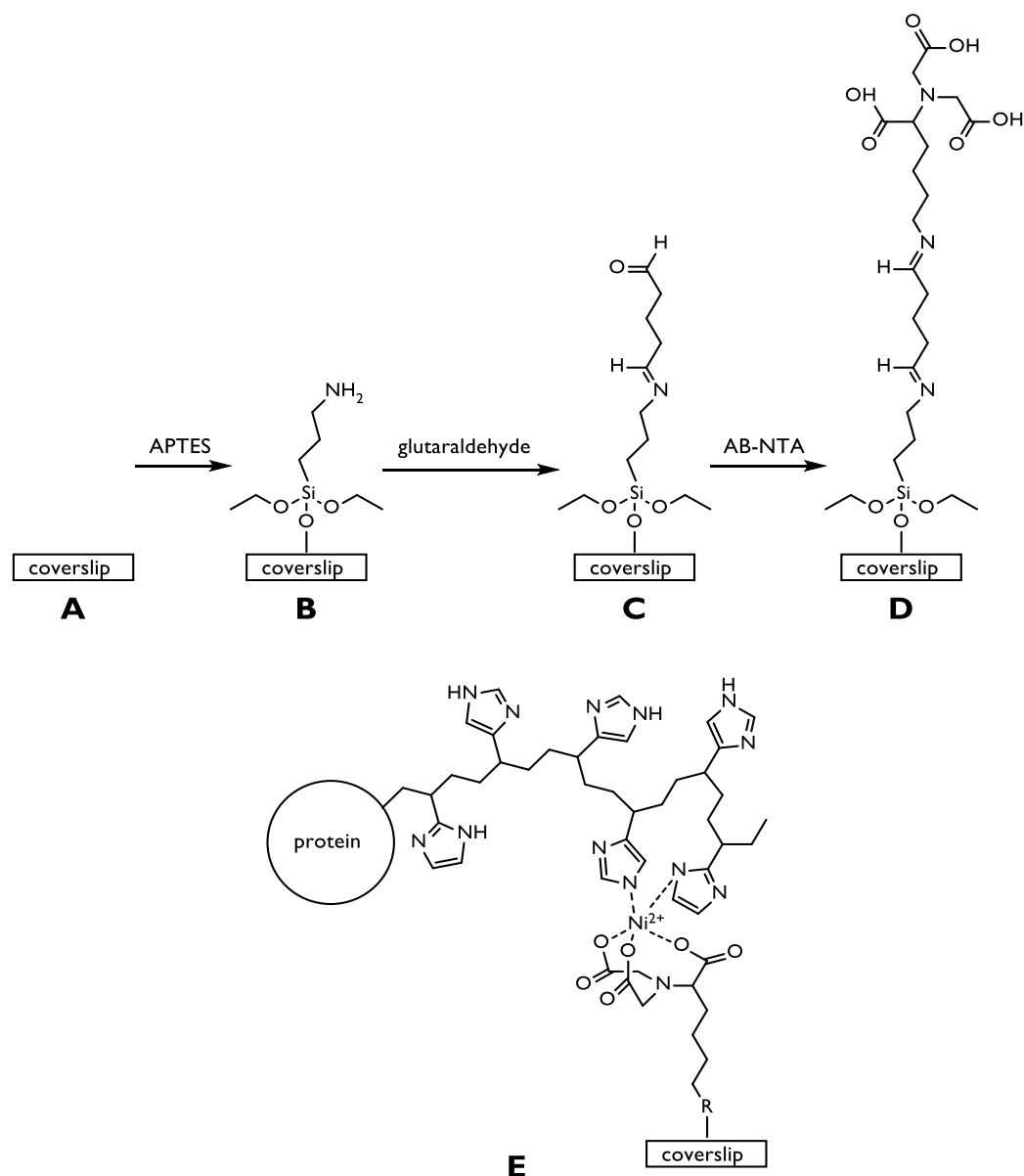


Figure 3.1. Schematic representation of surface modification.

The glass coverslip was cleaned with piranha solution and the first step of the RCA clean protocol prior to functionalization (A). (3-aminopropyl)triethoxysilane forms a covalently attached self-assembled monolayer on the substrate (B), which is modified by the addition of glutaraldehyde (C). Amino-labelled nitrilotriacetic acid can then couple to the added aldehyde groups via imine bond formation, yielding an NTA-terminated surface (D). When exposed to a solution of nickel(II) chloride, the NTA groups and nickel ions form a chelate which can then bind proteins through their histidine residues (E).

His-tagged protein subunits were site-specifically immobilized on microscope slide coverslips functionalized with nitrilotriacetic acid (NTA)[108]. The His_6 tags of the subunits bound to nickel ions complexed with the NTA; the same interaction was exploited for the protein purification via immobilized metal affinity chromatography (IMAC). Glass coverslips were modified through a series of chemical surface reactions performed on a precursor silane-based self-assembled monolayer (SAM). Aldehyde groups were coupled to the SAM

enabling amino-labelled NTA to react via imine bond formation and yield an NTA-terminated surface (see Figure 3.1).

3.2.1. Cleaning

The coverslip was cleaned and dried prior to functionalization. It was immersed in piranha solution (30 % hydrogen peroxide and 98 % sulphuric acid in the ratio 3:7) for at least 30 minutes then washed with copious amounts of water. It was subsequently subjected to the first step of the RCA clean protocol[109] (immersion in a mixture of water, ammonium hydroxide solution and 30 % hydrogen peroxide in the ratio 5:1:1 respectively, heated at 80 °C) for 30 minutes, rinsed with copious amounts of water, and dried in an oven at 80 °C overnight.

3.2.2. Surface modification

The coverslip was immersed in a solution of (3-aminopropyl)triethoxysilane (APTES, 40 mM in toluene) for 30 - 60 minutes. It was subsequently rinsed with toluene, then with a 1:1 mixture of toluene and ethanol, and finally with ethanol, and dried under a stream of nitrogen before immersion in a glutaraldehyde solution (12.5 % in water) for 20 – 30 minutes. Afterwards the coverslip was rinsed with water, then rinsed with acetone, and dried under a stream of nitrogen. Once dry, the coverslip was left overnight in a solution of N_{α},N_{α} -bis(carboxymethyl)-L-lysine hydrate (AB-NTA, 1 mM in water). This formed the functional layer of nitrilotriacetic acid, enabling protein immobilization as shown in Figure 3.1D – E. The coverslip was subsequently rinsed with water and dried under a stream of nitrogen. Unreacted aldehyde groups were blocked by immersing the coverslip in a solution of octylamine or decylamine (50 mM in ethanol) for 30 minutes. The functionalized coverslip was finally rinsed with ethanol and dried under a stream of nitrogen. It was cut into sections approximately 1 cm × 1.5 cm which were then stored in ethanol if not being used immediately.

3.2.3. Sample preparation

Flow chambers were constructed from functionalized coverslips and microscope slides as illustrated in Figure 3.2. A strip of clear double-sided tape slightly longer than the coverslip was applied along each long edge on one side of a microscope slide, leaving a gap 1 – 2 mm narrower than the coverslip in the centre. The tape was firmly pressed onto the slide using the side of a pipette tip. A coverslip section (dried under a stream of nitrogen from its storage ethanol) was then placed functionalized side-down in the centre of the microscope slide, its long sides supported by the strips of double-sided tape to produce a small chamber, open at either end, between the slide and the coverslip. A pipette tip was used to gently press the sides of the coverslip onto the tape. Rapid set epoxy adhesive was spread on either side of the open ends to create guiding channels for liquid. The epoxy adhesive also narrowed the chamber openings, spreading slightly by capillary action underneath the coverslip, which helped to slow evaporation from the chamber. The assembled flow chamber (approximately 10 μ l in volume) was left, protected from dust, for at least 15 minutes to ensure the epoxy adhesive was set before the application of liquids.

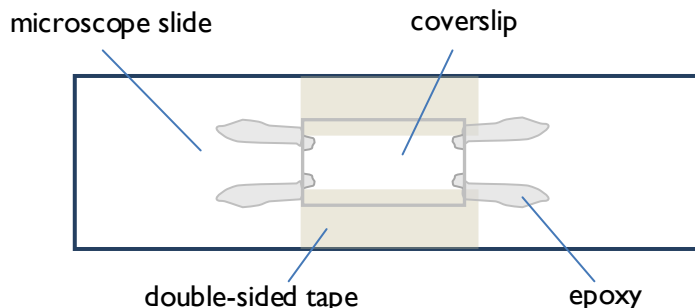


Figure 3.2. Schematic showing the construction of a microscope sample.

Double-sided tape increases the gap between the coverslip and the slide, allowing for the thicker buffer layer required for consistent operation of the microscope's Perfect Focus System[110].

Solutions were delivered to the flow chamber at one end via pipette and filled it by capillary action. A thin strip of filter paper applied to the other end wicked excess solution away while it was continuously supplied. Once wetted, the chamber was not allowed to dry out.

Unless stated otherwise, the following general procedure was used to prepare protein samples. The flow chamber was rinsed with 100 μl of water, and then with 100 μl of ~ 500 mM NiCl_2 which was left in the chamber for 5 minutes to charge the coverslip surface. The NiCl_2 solution was rinsed from the chamber with a further 100 μl of water before exchange into the appropriate buffer solution (100 μl rinse). 100 μl of the protein solution was then applied and incubated in the chamber for 5 minutes. Excess protein which had not bound to the coverslip was rinsed out with 100 μl of buffer solution. If the Gloxy oxygen scavenging system was to be used, 1 μl of 100 \times Gloxy solution was added to 99 μl of imaging buffer. This solution was then exchanged immediately into the sample chamber, just before imaging.

3.3. Optical arrangement for single molecule microscopy

3.3.1. Stoichiometry measurements

The optical bench layout used for experiments investigating the stoichiometry of the ChII-ChlD complex is shown in Figure 3.3. The apparatus consisted of a laser providing a source of excitation light; mirrors and lenses to guide, expand and focus the laser beam; an inverted fluorescence microscope to image the sample; and a camera (ORCA-Flash or Zyla sCMOS) to capture the images. The power of the laser beam was regulated either manually using a variable neutral density filter, or by computer control using the manufacturer's software (OBIS Connection, Coherent Inc.). The laser beam was expanded with a 3 \times beam expander before being focussed onto the back focal plane of a 60 \times oil immersion objective by a convex focussing lens. Within the microscope itself, a dichroic mirror reflected the excitation beam up to the objective and the sample, and allowed the returning fluorescence emission to pass through to a filter which removed any stray wavelengths. The filtered emission signal was then directed to the camera for recording. A schematic of the illumination configuration within the microscope is given in Figure 3.4 A.

When mounted on a second linear translation stage with displacement perpendicular to the direction of propagation of the laser beam, the focussing lens enabled the use of TIRFM via the objective lens method (illustrated in Figure 3.4 B). Translating the focussing lens in a plane perpendicular to the direction of the laser beam caused the beam to propagate at an angle through the objective lens, and thus to meet the boundary between the coverslip and the buffer of the sample at an angle. If the focussing lens was translated far enough, the angle of incidence at this boundary became greater than the critical angle and complete total internal reflection occurred at the glass-buffer interface.

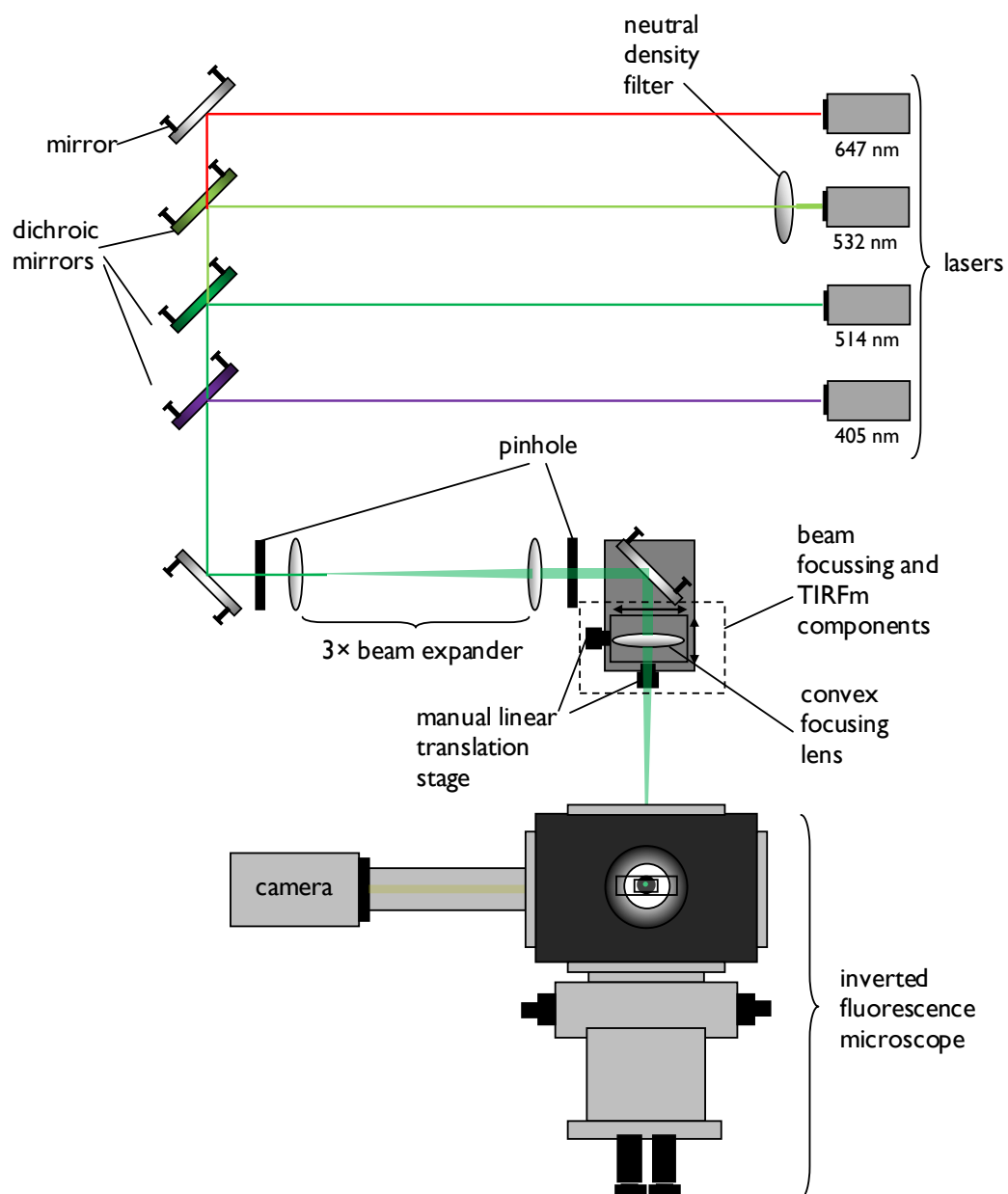


Figure 3.3. Schematic of the optical bench setup used in experiments investigating the stoichiometry of the ChII-ChlD complex.

Neutral density filters were employed to control the power of those lasers which were not adjustable.

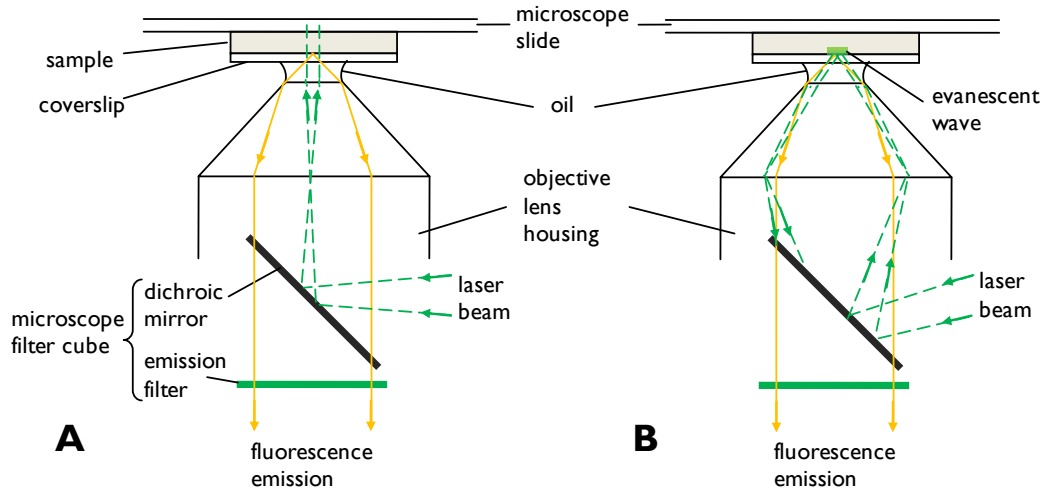


Figure 3.4. Schematic showing the illumination configuration within the microscope in (A) epi-fluorescence mode and (B) TIR mode.

In epi-fluorescence mode the incident laser beam passes through the centre of the microscope objective and emerges perpendicular to the coverslip/buffer interface. In TIR mode the laser beam passes through the periphery of the objective corresponding to higher values of numerical aperture so that it emerges at an angle greater than the critical angle required to achieve total internal reflection at the coverslip/buffer interface.

Transition into complete TIR mode was ensured as follows. The laser beam had previously been correctly aligned in epi-fluorescence mode by an iterative process of alternately adjusting the two mirrors just upstream from the microscope to direct the beam simultaneously through the centres of the rear and front focal planes of the objective. Correct alignment was further verified by observing the image of a fluorescent sample and confirming that it defocused symmetrically about the x and y axes when alternating between focussed and defocused images. A sample of immobilized TetraSpeck beads against a background of beads free in solution was used to track the progress of the system as TIR mode was entered. The TetraSpeck sample was viewed via the camera as the focussing lens was translated sideways in the plane perpendicular to the beam direction. The fluorescence background from the freely diffusing beads in the image gradually decreased and the images of the immobile beads became brighter as the incident angle of the laser beam increased. The focussing lens was translated in one direction until the fluorescence image disappeared and was then translated back again in the opposite direction until the fluorescence image just reappeared. This corresponded to the lowest possible penetration depth of the evanescent illumination field for the system.

The critical angle for the system can be calculated using Equation 1.3 with $n_t = 1.33$ for aqueous buffer and $n_i = 1.52$ for glass, giving $\theta_c = 61.0^\circ$. As long as the laser beam meets the coverslip/buffer interface at an angle greater than 61° , complete total internal reflection will occur. The numerical aperture of the objective not only determines the angular spread over which it can gather light from a sample, but also the angular spread over which it can deliver light to a sample. The maximum angle of incidence which the objective can deliver to the sample is calculated using Equation 1.5 with $NA = 1.49$ and $n_{oil} = 1.52$, giving $\theta_m = 78.6^\circ$. If this maximum angle is exceeded, the laser beam will not exit the objective and

reach the sample, hence the observation that the fluorescence image disappeared when the focussing lens was translated past a certain point (approximately 5 mm off centre). By translating the focussing lens back from the point at which the fluorescence image was lost, until the fluorescence image was just re-established, an angle of incidence close to the maximum possible was attained. According to Equation 1.4 which gives the relationship between intensity and penetration depth for an evanescent field, the minimum penetration depth d (corresponding to the maximum possible angle of incidence) possible for this system under 532 nm illumination is 63 nm.

When the sample structure allowed it, the microscope's Perfect Focus System (PFS) [110,111] was used to automatically compensate for drift in the z-axis (perpendicular to the plane of the sample) during imaging. The PFS uses an 870 nm infrared light to monitor the position of the coverslip/buffer interface in the sample, and adjusts the vertical position of the objective lens to maintain a constant focal distance. It consists of a near-IR LED, a dichromatic mirror, an offset lens and a CCD line sensor housed in a specialized nosepiece unit which is separate from the primary microscope optical train, allowing the PFS to be engaged when required without interfering with fluorescence imaging. When the PFS is engaged the dichromatic mirror is introduced into the beam path between the tube lens and the objective where it reflects the IR beam from the LED towards the sample but allows transmission of other wavelengths used in fluorescence imaging. The IR beam is projected through the periphery of the objective so that when the objective lens is focussed on the coverslip/buffer interface (the reference plane), the IR beam undergoes total internal reflection. The returning IR beam is reflected by the dichromatic mirror towards the centre of the CCD line sensor. If focal drift occurs the reference plane moves in relation to the objective lens, changing the angle at which the returning IR beam is reflected from the dichroic mirror, and causing the beam to hit the CCD off-centre. The vertical position of the objective lens is then adjusted to bring the reflected IR beam back to the centre of the CCD sensor. Once the reference plane has been found and TIR of the IR beam is established, the focal plane of the objective can be adjusted to view the sample by applying an offset. The offset is implemented with the offset lens which both directs the incident light from the LED to the dichroic mirror and guides the returning reflected light to the CCD line sensor. When an offset is applied the position of the offset lens is shifted in the plane of propagation of the IR beam to change the distance between the lens and the CCD. This allows the objective lens to focus on the sample while the IR beam is reflected from the reference plane and still hits the centre of the CCD sensor.

In order to avoid frustrated total internal reflection of the IR beam within the sample, the buffer layer must be at least several wavelengths in thickness. This means that samples constructed simply from a coverslip sealed to a microscope slide are generally not suitable for use with the PFS as the evanescent field generated by the IR beam at the coverslip/buffer interface extends into the glass of the microscope slide on the other side of the buffer layer with appreciable amplitude, transferring energy and enabling the beam to propagate onwards. The buffer layer in the sample chamber illustrated in Figure 3.2 is approximately 100 μm in thickness, which is over 100 times the wavelength of the IR illumination, and therefore capable of supporting operation of the PFS.

The sources of excitation illumination for TNP, Cy3 and Cy5 dyes were 405 nm, 532 nm and 647 nm CW lasers respectively. The filter cube components used for experiments involving Cy5 and TNP dyes, or TNP dye only, were a 545/650 nm dual-edge dichroic and a 488/647 nm dual-notch emission filter. For Cy5 only the standard Cy5 filter cube was employed. The filter cube components used for experiments involving Cy3 were a 545/650 nm dual-edge dichroic and a 565/24 nm bandpass emission filter.

3.3.2. Nucleotide binding measurements

The optical bench layout used in experiments investigating nucleotide binding to ChII is shown in Figure 3.5. The apparatus consisted of a 514 nm laser providing excitation illumination; mirrors and an optical cable to guide the laser beam; an inverted fluorescence microscope with additional TIRF module to image the sample; and a water-cooled Zyla sCMOS camera to capture the images. The power supply of the laser provided variable power control, and the beam could be further attenuated by introducing neutral density filters into the beam path within the Ti-LAPP System. The laser beam was expanded and focussed onto the back focal plane of the microscope objective (a 60× or 100× oil immersion lens) within the Ti-LAPP System. Within the microscope itself, a 532 nm single-edge dichroic mirror reflected the excitation beam up to the objective and the sample, and allowed the returning fluorescence emission to pass through to a 565/24 nm bandpass emission filter which removed any stray wavelengths. The filtered emission signal was then directed to the camera for recording.

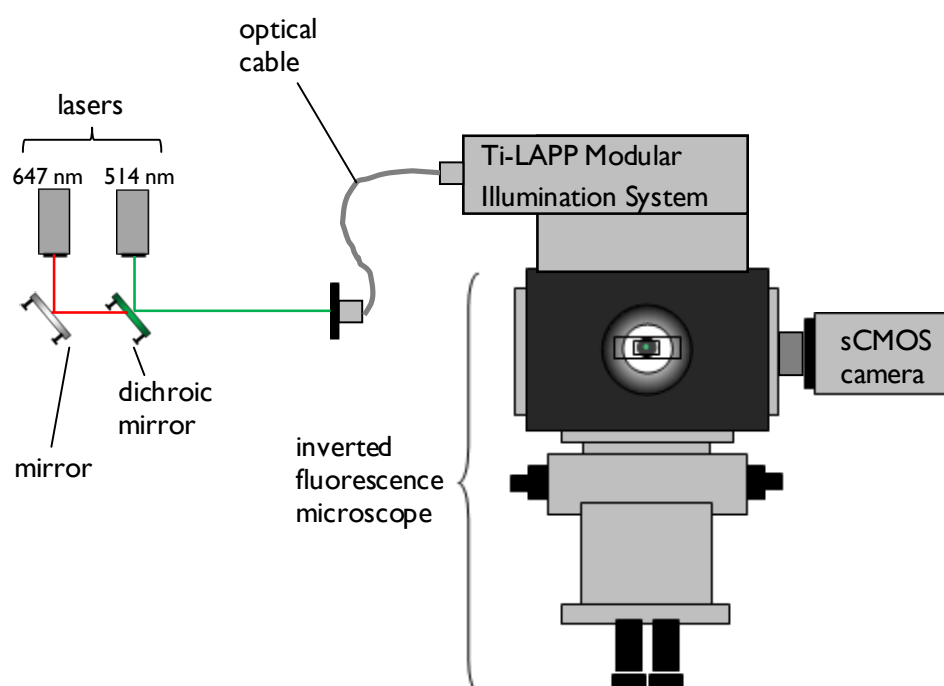


Figure 3.5. Schematic of the optical bench layout for experiments investigating nucleotide binding to ChII.

Entry into TIRFM mode by the objective method followed the same principles as outlined in Section 3.3.1. and illustrated in Figure 3.4. The displacement required for TIR was recorded by the Ti-LAPP System TIRF module so that switching between epi-fluorescent and TIR

mode was automated. The microscope's PFS was used to automatically compensate for drift in the z-axis (perpendicular to the plane of the sample) during imaging.

3.4. Data acquisition

The lasers, microscope shutter and cameras were operated independently. The cameras were controlled using the manufacturer's software (HCLImage Live (2013), Hamamatsu Photonics UK Limited, Welwyn Garden City, UK for the ORCA-Flash; Solis I Acquisition and Analysis Software, Andor Technology Limited, Belfast, UK for the Zyla); this was also used for recording of the image series. Details of camera operation (frame rates, series lengths etc.) are given in the relevant results sections. Images were saved after recording for later analysis.

4. Analysis of microscope images

TIF format image sequences were processed using the standard tools provided with the free image processing package Fiji[112], and analysed using custom functions written with the commercial software package MATLAB (MATLAB Student R2011a, The MathWorks Inc., Natick, US). The methods used in image analysis are detailed in the following sections. Each section contains a figure summarizing the steps involved, accompanied by an explanation in the main text. When the term ‘fluorophore’ is used in the context of stepwise photobleaching data, unless otherwise stated this refers to the group of 1 or more fluorescent dye molecules labelling a protein complex, which appear as a single fluorescent spot due to their sub-diffraction limit separation. In the context of fluorescent nucleotide binding the general term ‘fluorophore’ refers to a stationary fluorescent spot identified as a nucleotide-binding complex (the fluorescence of which is due to one or more fluorescent nucleotide molecules). The MATLAB code described here can be found in Appendices A to C with explanations of function arguments.

4.1. Localization of fluorophores

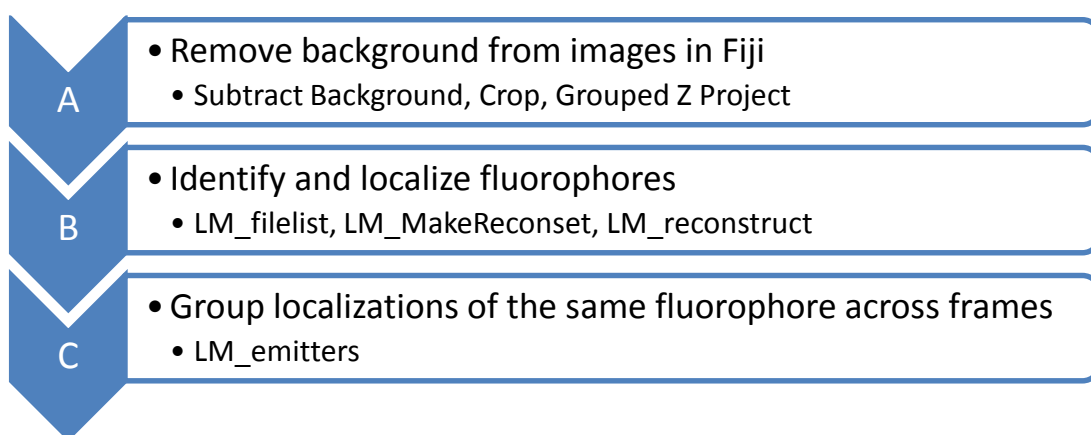


Figure 4.1. Summary of steps required to localize fluorophores within an image series.

Unless explicitly stated otherwise, analysis was carried out in MATLAB. The tools/functions used in each step are listed below the description.

Custom MATLAB code described in Section 4.1. (known collectively as LM_v8) was written by Robert Turner (MBB, University of Sheffield) and was used as a base for the present work[113]. It uses photoactivation localization microscopy/STORM methodology as described previously by Betzig et al.[114]

A. Remove background from images in Fiji

Image series were processed in Fiji as virtual stacks. The Virtual Stack command within the Process>Batch menu was used to implement the Subtract Background and Crop tools. Subtract Background removed the background from each frame of an image series individually using a rolling ball algorithm with a radius of 10 pixels (Figure 4.2). If needed, the images in the series were cropped to include only the centre of the illuminated area. In the case of nucleotide binding studies, the images in a series were summed using the Grouped Z Project command in the Image>Stacks>Tools menu to create a new image series

in which each image was the sum of 10 to 100 of the original images. The summed image series was then used in fluorophore identification and localization.

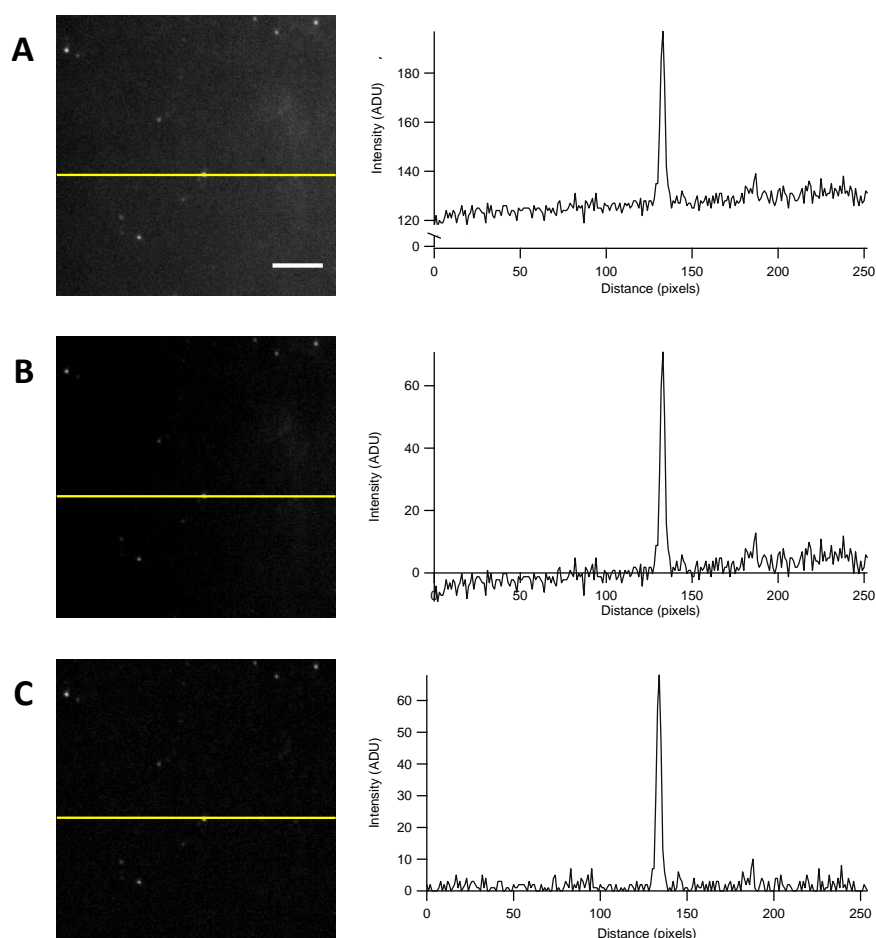


Figure 4.2 Background removal.

Comparison of an image before and after background removal with accompanying intensity plot from the linescan indicated in yellow. (A) Original image; (B) image after subtraction of the mean intensity value (negative values are displayed as zero i.e. black); (C) image after rolling ball background subtraction with a radius of 10 pixels. The average deviation of the background from zero is 127 ADU/pixel, 3.98 ADU/pixel, and 2.34 ADU/pixel for images A, B and C respectively. This shows that the rolling ball method is more effective for background removal. Bar = 5 μm . Molecules are His₆-ChlD-Cy3 in the presence of 100 nM Cy3-ATP, viewed under 532 nm illumination with 0.01 s exposure in TIR mode.

Rolling ball background removal was employed for its ability to cope with uneven background variations. It works by subtracting a locally calculated background intensity from each pixel, which is determined from the average of the surrounding pixels contained within the specified radius. It can be visualized as rolling a ball over the underside of the image, where intensity is equivalent to depth. The ball will follow the contours of the background and any features which are larger than the radius of the ball, but pass over smaller features (such as single molecules). The ball traces out the background of the image which is then subtracted to reduce the background to zero while the intensities of single molecules are retained. Figure 4.2 compares rolling ball background subtraction to average

background correction, where the mean background value is subtracted from the image. In the case of an uneven background, the rolling ball method performs better, producing a flat background (mean deviation from zero of 2.34 ADU/pixel) whereas the average background correction retains the shape of the uneven background (mean deviation from zero of 3.98 ADU/pixel).

B. Identify and localize fluorophores

MATLAB was used for all subsequent image analysis. A list of the file paths to individual image files in a series was retrieved and stored in a cell array using the function LM_filelist. The parameters used in identifying potential fluorophores (threshold brightness, fluorophore spacing etc.) were specified and stored in a structure array (a MATLAB data type enabling grouping of related data) via the function LM_MakeReconset. Fluorophores which registered above the threshold brightness specified in the parameters structure were identified and localized in every frame of the image series using the function LM_reconstruct (Figure 4.3).

LM_reconstruct passed each image in the series through a Laplacian of Gaussian filter with standard deviation equal to 1 pixel and then created a binary mask of regions within the filtered image which had intensity values greater than the specified threshold value. The weighted centroid of each region was calculated from the intensity values in the original image to provide initial x- and y-coordinates for each fluorophore. (The z-direction is taken as being the axial direction, perpendicular to the sample plane). A square section (the dimensions of which were specified in the parameters structure), centred on the weighted centroid position for each fluorophore, was extracted from the original image and used to perform a 2-dimensional Gaussian fit on each fluorophore. This fit returned the planar position and dimensions of the fluorophore at sub-pixel resolution.

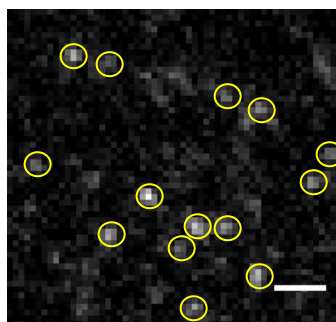


Figure 4.3. Fluorophore localization.

Yellow circles indicate identified and localized fluorophores. Bar = 1 μm . Molecules are His₆-ChlD-Cy5 immobilized with ChlI at 25 pM in the presence of 10 μM TNP-ATP, viewed under 647 nm illumination with 0.1 s exposure in TIR mode, the image being the sum of 10 individual frames.

C. Group localizations of the same fluorophore across frames

The previous analysis step created a matrix containing information on all fluorophore localizations across all frames in the image series. The function LM_emitters was then used to link localizations of the same fluorophore between frames by assigning a number identifier to every fluorophore and storing it in the matrix. Two localizations in any two

frames were considered as being the same fluorophore (and assigned the same identifier) if they occurred within 2 pixels of each other in both the x- and y-directions. Thus, the position of a particular fluorophore was determined in every frame (Figure 4.4).

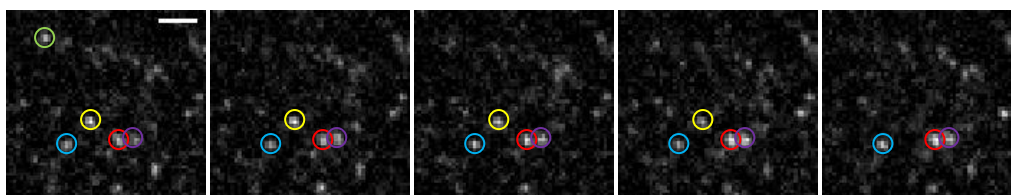


Figure 4.4. Fluorophores are tracked through every frame of an image series.

The colours of the circles represent different identifier numbers assigned to each fluorophore, enabling them to be tracked in each frame. Bar = 1 μm . Molecules are His₆-ChID-Cy5 immobilized with ChII at 25 pM in the presence of 10 μM TNP-ATP, viewed under 647 nm illumination with 0.1 s exposure in TIR mode, each image being the sum of 10 individual frames.

4.2. Stepwise photobleaching analysis for stoichiometry

4.2.1. Refinement of fluorophore list and intensity extraction

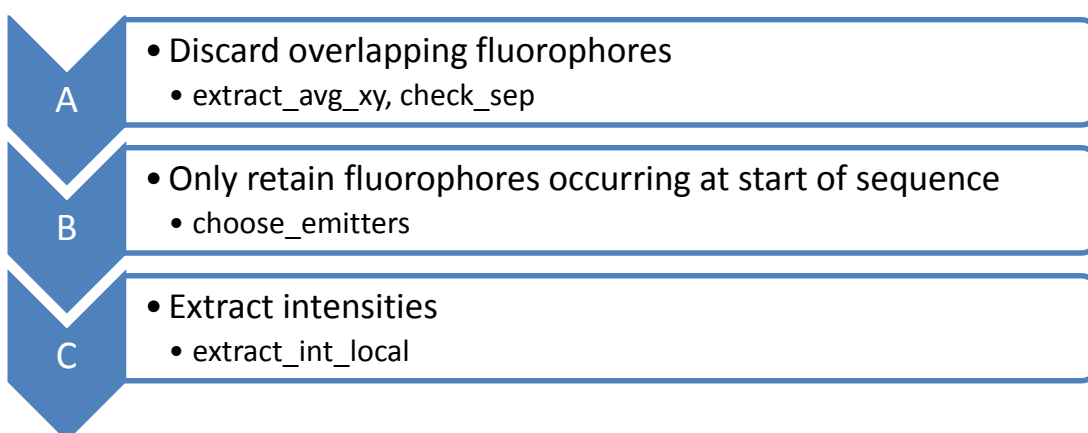


Figure 4.5. Summary of steps required to refine the list of localized fluorophores and extract their intensities over an image series. The custom MATLAB functions used in each step are listed below the description.

A. Discard overlapping fluorophores

The matrix of fluorophore data generated as described in Section 4.1. was refined to remove fluorophores found to occur closer to each other than a user-specified distance (deemed to be overlapping, see Figure 4.6). In the cases where localization was only performed on one frame in the series, or where drift was small enough to allow the use of average x- and y-positions, the average x- and y-coordinates for each fluorophore were extracted from the matrix of fluorophore data using the function `extract_avg_xy`. The function `check_sep` was then employed to identify and discard overlapping fluorophores based on the average x- and y-positions if possible, or else by comparing each set of localizations frame by frame. The relative pixel size in an image was calculated by dividing the physical pixel size (6.5 μm) by the magnification of the objective used since no lenses were placed between the objective and the camera.

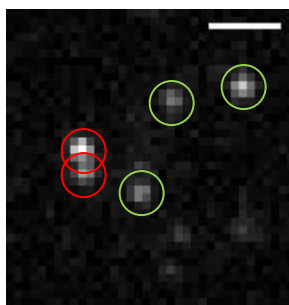


Figure 4.6. Overlapping fluorophores are discarded.

Fluorophores found to overlap by more than a user-specified distance were excluded from further analysis. The fluorophores circled in red were excluded from the dataset. Bar = 1 μm . Molecules are His₆-ChlD-Cy5 immobilized with ChlI at 25 pM in the presence of 10 μM TNP-ATP, viewed under 647 nm illumination with 0.1 s exposure in TIR mode, the image being the sum of 50 individual frames.

B. Only retain fluorophores occurring at start of sequence

In order that only complexes which underwent stepwise photobleaching were included in analysis, the fluorophore list was further refined using the function `choose_emitters`. Only those fluorophores which registered in the first 1 s of the image sequence were retained.

C. Extract intensities

The function `extract_int_local` was used to retrieve the intensities of the fluorophores in the refined list for every frame in the image series (Figure 4.7). The individual frame localizations for each fluorophore were used, allowing correction for any drift which occurred during image acquisition (generally no more than 300 nm). In frames where a fluorophore had not been detected, its current position was inferred from its previous or next detected position. Intensities were determined by taking the sum of the pixel values in a square area surrounding the pixel containing the localized fluorophore position, with sides roughly equal to the diameter of the fluorophore point spread function (PSF). The fluorophore PSF is the image of the point-like fluorophore produced by the diffraction-limited optics of the microscope system. The fluorophore, having dimensions on the order of 1 nm, is much smaller than the wavelength of light used to detect it, so the size of its image or PSF depends only on the wavelength of light which it emits and the objective lens used to view it as described by Equation 1.1.

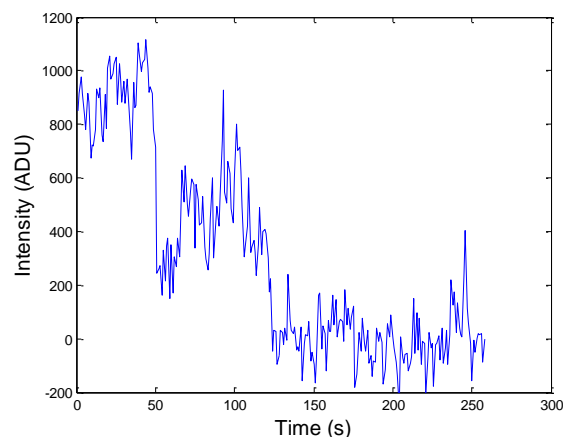


Figure 4.7. Sample photobleaching steps in the intensity-time trace of a fluorescently labelled protein complex.

The protein is His₆-ChID-Cy3 with ChII in the presence of 5 mM ATP, viewed under 532 nm illumination with 1 s exposure.

The average intensity of a single fluorophore was determined from samples of fluorescently labelled His₆-ChID denatured in 8 M urea to prevent subunit association. The fluorophore intensity was calculated from the difference in average intensity before and after a bleaching event. Figure 4.8 shows the distribution of single molecule fluorescence values for Cy3-labelled ChID molecules. A Gaussian fit to the distribution indicates a mean step size of 408 ± 55 ADU, which agrees with the step sizes shown in Figure 4.7 for a Cy3-labelled ID complex.

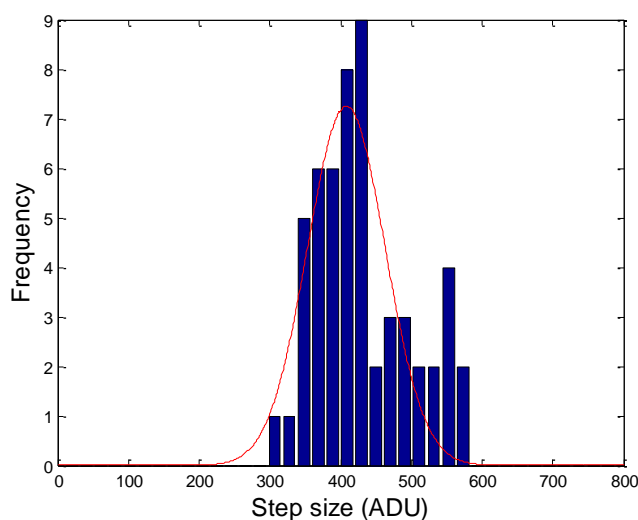


Figure 4.8. Determination of single fluorophore intensity.

Distribution of intensities of single Cy3 molecules viewed under 532 nm illumination with 1 s exposure. The fitted Gaussian distribution indicates a mean step size \pm standard deviation of (408 ± 55) ADU.

4.2.2. Determination of complex stoichiometry

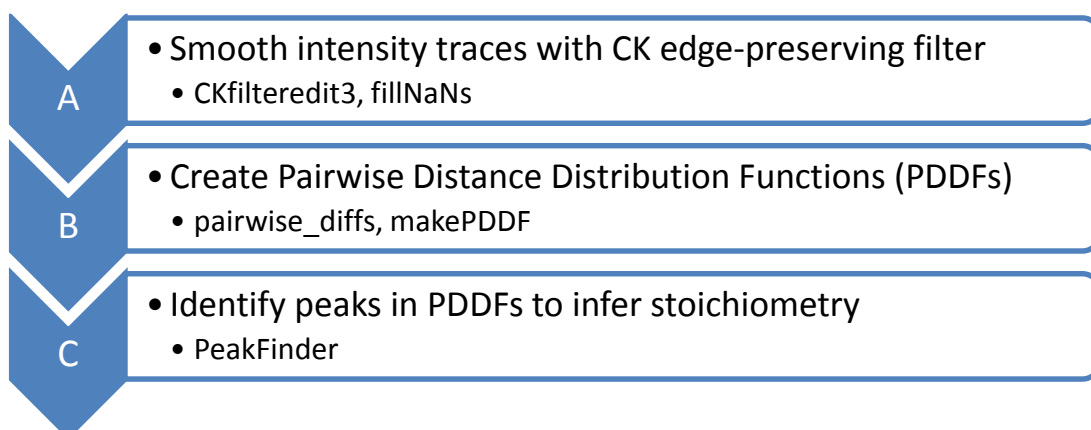


Figure 4.9. Summary of steps required to infer stoichiometry from stepwise photobleaching intensity traces. The custom MATLAB functions used in each step are listed below the description.

A. Smooth intensity traces with CK edge-preserving filter

Before attempting to discern different intensity levels in the intensity traces for each fluorophore, the traces had to be smoothed to aid automatic detection. The Chung Kennedy (CK) filter[77] was chosen for this purpose as it is better able to preserve the edges of step-like transitions than previously widely used low pass and averaging filtering techniques. It is a non-linear digital filtering technique using forward and backward processing to avoid averaging over step changes in a signal.

Fluorophore intensity traces were processed 3 times with the CK filter function CKfilteredit3, using a bank of 5 predictors with lengths of 4, 8, 16, 32, and 64 data points; an analysis window of between 4 and 10 data points; and a p weighting factor of 10 (Figure 4.10). All predictors were treated as equally probable. After each filtering pass, the smoothed intensity traces were checked using the function fillNaNs. Occasionally, 2 or more consecutive data points have the same intensity value. The CK filter fails when it encounters these points, returning NaN values in the data. The possibility of this occurring becomes greater with each CK filter pass as noise in the intensity trace is smoothed out. The fillNaNs function checks the intensity traces after each CK filter pass, and if it encounters a NaN value it substitutes a value inferred from the preceding data point. This was mostly required after the third and final CK filter pass.

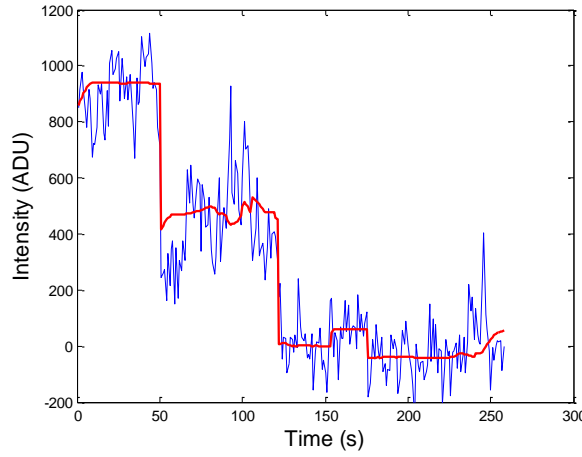


Figure 4.10. The Chung Kennedy (CK) filter smooths noise in the intensity trace while preserving the edges of abrupt transitions in intensity level.

The original intensity trace (blue) is overlaid with the CK filtered trace (red). The protein is His₆-ChID-Cy3 with ChII in the presence of 5 mM ATP, viewed under 532 nm illumination with 1 s exposure.

B. Create Pairwise Distance Distribution Functions (PDDFs)

Steps in the intensity traces were detected using an adaptation of the pairwise distance distribution and power spectrum method described by Svoboda et al.[70] to determine average step size in studies of kinesin movement. To produce a pairwise distance distribution function (PDDF) the difference in intensity, ΔI_{ij} , for all data point pairs in an intensity trace $I(t)$ containing n data points is calculated:

$$\Delta I_{ij} = I(t_j) - I(t_i) \quad \text{where } i = 1:n, \quad j = 2:n, \quad j > i$$

Equation 4.1. Calculation of pairwise differences in stepwise photobleaching intensity traces

The difference values are then arranged into a histogram (Figure 4.11).

A PDDF was calculated for each intensity trace, but instead of then calculating the one-sided power spectrum of the PDDF to determine step size, the peaks in the PDDF were identified and fitted directly. This was done for two reasons. Firstly, unlike molecular motor stepping data which contains many tens of similarly sized steps, stepwise photobleaching stoichiometry data generally contains less than 10 steps; in fact, the maximum number of photobleaching steps which have been observed directly without resorting to mathematical extrapolation techniques is 12 [68]. Power spectra highlight the periodicity present in a PDDF, and so work best with greater numbers of steps. Secondly, the power spectrum analysis only returns an average step size while information on the number of steps present is lost.

Pairwise differences were calculated for each intensity trace using the function `pairwise_diffs`. The pairwise differences were then arranged into histograms of 2000 bins each, using the function `makePDDF` to produce PDDFs for all fluorophores at once (Figure 4.11). The positions of the peaks in the PDDF indicate multiples of the intensity value of the single step size, and the number of peaks is equal to the number of photobleaching steps

(excluding the zero-centred peak). The largest peak occurs around 0 because many of the pairwise differences are calculated between elements having similar intensities. The width of the zero-centred peak then gives an estimate of the noise in the filtered signal.

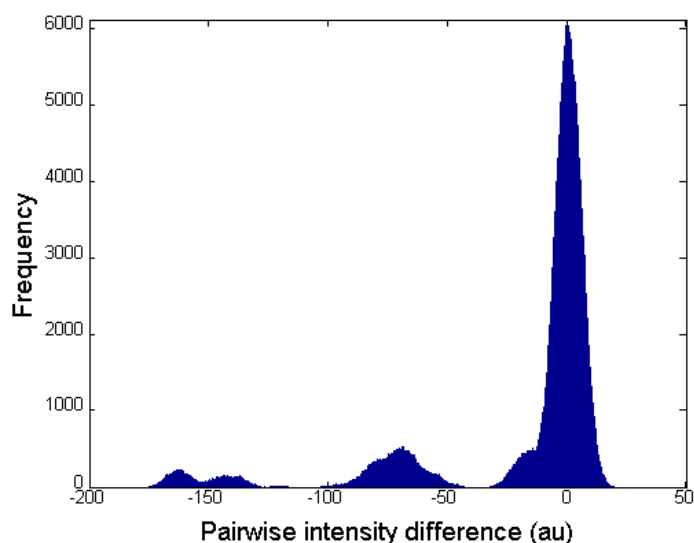


Figure 4.11. Sample pairwise distance distribution function highlights the discrete intensity levels in the stepwise photobleaching trace of a fluorescently labelled protein complex.

The protein is His₆-ChlD-Cy5 with ChlI in the presence of 5 mM ATP, viewed under 647 nm illumination with 0.2 s exposure.

C. Identify peaks in PDDFs to infer stoichiometry

PDDFs were smoothed with MATLAB's Savitzky-Golay filter using a first degree polynomial and a frame size of 99 points. The function PeakFinder was then employed to identify and extract information on the peaks in each PDDF (Figure 4.12). It adapts and expands upon a peak flagging routine written by Nigel Reuel[76]. Given parameters determining the minimum size and separation expected for a peak to be considered significant, the function further smooths the PDDF with a short running average filter if necessary and then flags up potential peaks. It compares the separation between adjacent peaks, and between peaks and neighbouring valleys, to determine how to classify each potential peak. Once a final list of peaks is settled on, the peak positions are determined from individual Gaussian fits. The Gaussian fits provide an estimate of the noise associated with the data, and another opportunity to reject any false peaks having too large a standard deviation. The function returns a vector containing the peak numbers detected for each intensity trace, and matrices containing the details of the peak fits associated with steps and with the zero-centred peak for each trace.

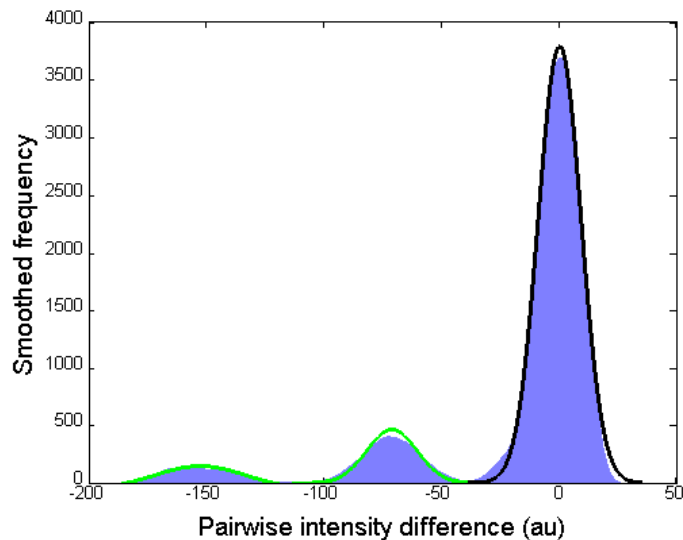


Figure 4.12. Peak identification in a pairwise distance distribution function (PDDF).

The smoothed PDDF is overlaid with the Gaussian fits to the data. The peak corresponding to the background noise is shown in black. The protein is His₆-ChlD-Cy5 with ChlI in the presence of 5 mM ATP, viewed under 647 nm illumination with 0.2 s exposure.

4.3. 'On' and 'off' time analysis for fluorescent nucleotide binding

4.3.1. Refinement of fluorophore list and intensity extraction

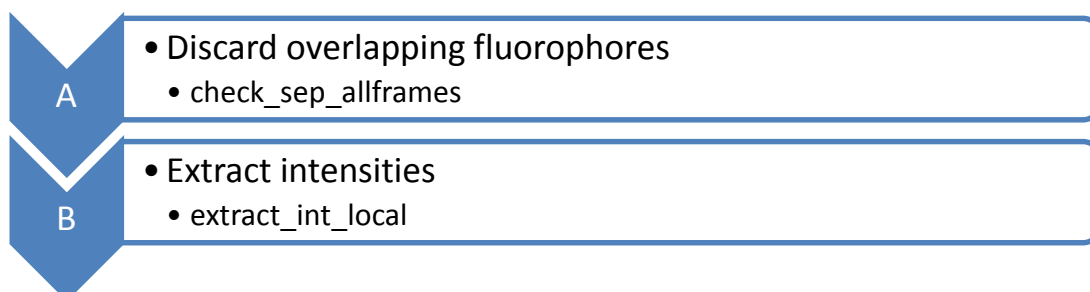


Figure 4.13. Summary of steps required to refine the list of localized fluorophores and extract their intensities over an image series. The custom MATLAB functions used in each step are listed below the description.

A. Discard overlapping fluorophores

The matrix of fluorophore data generated as described in Section 4.1. was refined to remove fluorophores found to occur closer to each other than a user-specified distance (deemed to be overlapping). Only those image series in which drift was small enough to allow the use of average x- and y-positions across a whole sequence were used. The function `check_sep_allframes` was used to calculate the average x- and y-coordinates for each fluorophore, and then to identify and discard overlapping fluorophores based on these average positions.

B. Extract intensities

The function `extract_int_local` was used to retrieve the intensities of the fluorophores in the refined list for every frame in the original, un-summed image series (Figure 4.14). The

fluorophore localization for each frame in the summed image series was assigned to the corresponding 100-frame block in the original series. In frames where a fluorophore had not been detected, its current position was inferred from its previous or next detected position. Intensities were determined by taking the sum of the pixel values in a square area surrounding the pixel containing the localized fluorophore position, with sides approximately equal to the radius of the fluorophore PSF.

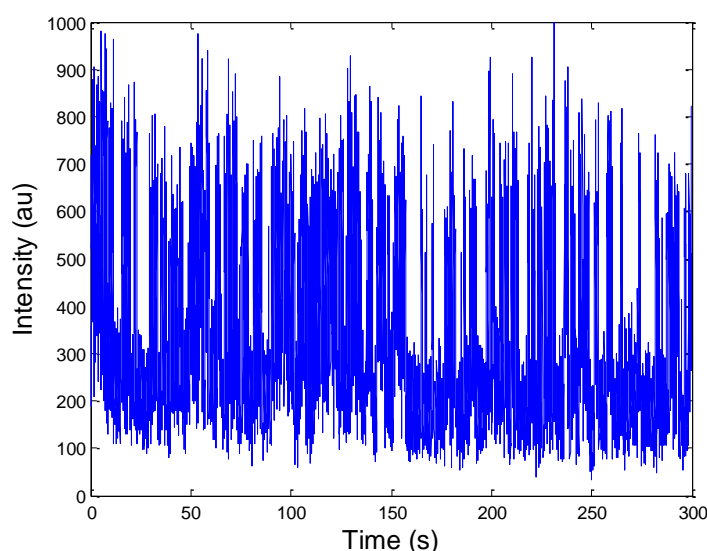


Figure 4.14. Intensity as a function of time for fluorescently labelled nucleotide binding to and dissociating from a ChII molecule.

High intensity levels correspond to nucleotide binding events. The protein is His₆-ChII in high salt buffer with 60 nM Cy3-ATP, viewed under 514 nm illumination with 0.01 s exposure.

4.3.2. Extraction of event durations and amplitudes

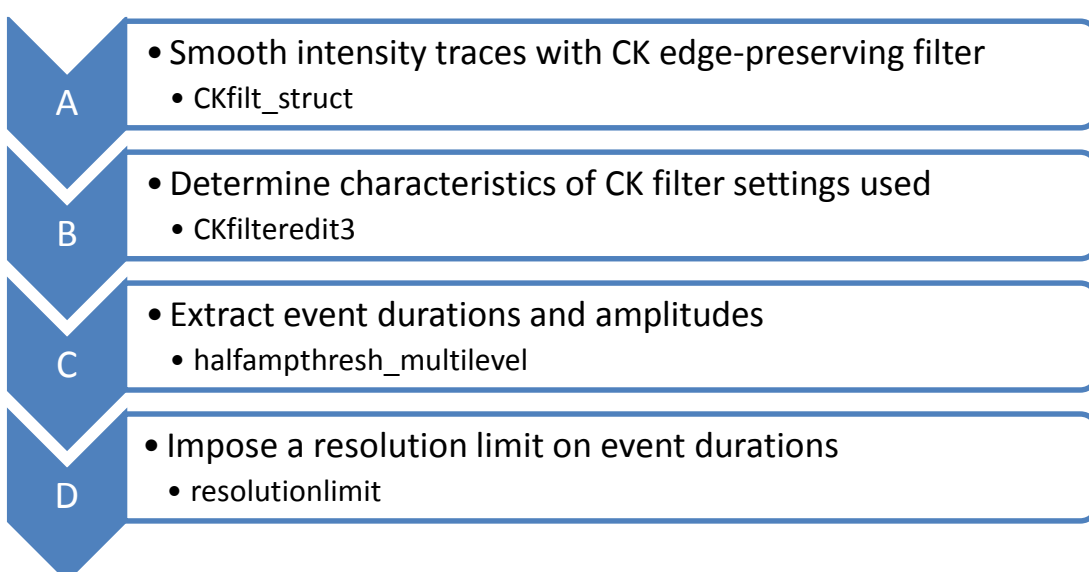


Figure 4.15 Summary of steps required to extract binding event durations and their amplitudes over an image series. The custom MATLAB functions used in each step are listed below the description.

A. Smooth intensity traces with CK edge-preserving filter

The information gathered for each fluorophore was ordered into an array of structures with separate fields for each attribute such as fluorophore number, intensity over time etc. for ease of manipulation. The full description of the structure fields is given in Appendix C. The intensity traces were smoothed using a Chung Kennedy filter with a bank of 4 predictors with lengths of 2, 4, 8, and 16 data points; an analysis window of 5 data points; and a p weighting factor of 50 (Figure 4.16). All predictors were treated as equally probable. This increased the signal to noise ratio to around 5 and reduced the rate of false event detection to an average of 0.002 per frame (this rate varied between individual intensity traces).

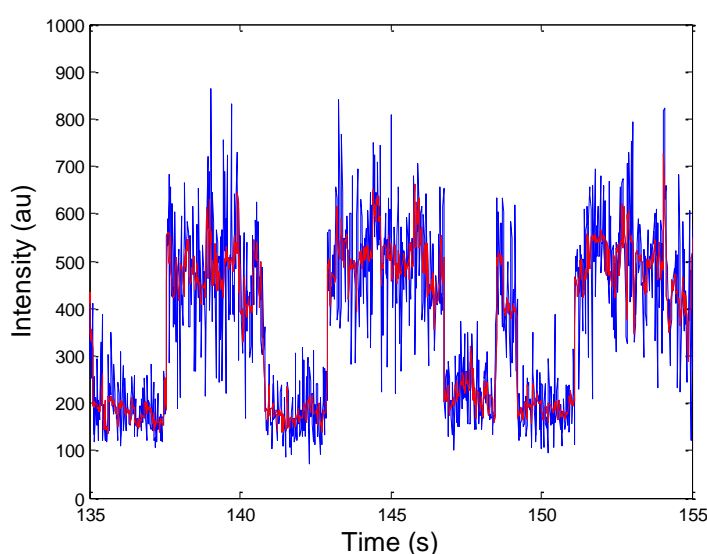


Figure 4.16. Chung Kennedy (CK) filtering increases the signal to noise ratio and reduces the number of false events detected.

The original intensity-time trace (blue) is overlaid with the CK filtered trace (red). The trace shows the change in fluorescent signal as Cy3-ATP binds to and dissociates from a ChII molecule (His₆-ChII in high salt buffer with 60 nM Cy3-ATP, viewed under 514 nm illumination with 0.01 s exposure).

B. Determine characteristics of CK filter settings used

While smoothing the intensity traces improves signal to noise and reduces the probability of detecting false events (instantaneous fluctuations of the intensity signal large enough to cross a threshold value due to noise rather than fluorescent nucleotide), it also affects the apparent durations of detected events (on or off periods in the intensity trace). A Chung Kennedy filter helps to preserve the edges of step changes in intensity level better than averaging filters, but it still shortens the apparent length of events with durations comparable to or shorter than its averaging window length. If a true picture of the distribution of event durations is to be obtained, corrections must be made for the behaviour of the smoothing filter.

The CK filter characteristics were determined in a similar fashion to that outlined by Colquhoun and Sigworth[96] for a Gaussian filter. The CK filter response was evaluated for the settings used to filter the intensity traces with respect to filtered event width (duration in frames), false event detection rate, and filtered event amplitude. In order to generate

simulated data for the evaluation, the signal to noise ratio of the sample intensity traces was determined. CK filtered intensities were sorted into histograms, allowing easy identification of intensity levels using the peak-finding method described previously for stoichiometry determination (Section 4.2.2.). The standard deviation of the noise was determined by sampling the background intensity level of unfiltered intensity traces.

Simulated data were generated using MATLAB. An intensity trace of 15000 data points in length was produced by setting all values to 0 initially. A single unit-amplitude event was created by setting the desired number of data points in the centre of the trace to a value of 1. Noise was introduced to the trace by adding normally distributed noise with a mean of 0 and standard deviation proportional to that of the sample data. The width of the unit-amplitude event was varied between 1 and 20 data points. A half-amplitude threshold approach was adopted for event detection (for real as well as for simulated data) as this means that both on and off event durations can be determined without bias. The threshold level was set to be half of the on event amplitude and every crossing of the threshold was interpreted as a change between the on and off states. Simulated data were smoothed with the CK filter using the same settings as for the sample data, and the final event duration (w_t) and amplitude (A) were determined. The number of false events was estimated by performing half-amplitude threshold analysis in the negative direction on the stretches of zero-level intensity within the simulated intensity traces, giving a false event rate of $(1.6 \pm 0.4) \times 10^{-3}$ per frame for a signal to noise ratio of 3.

Final event durations and amplitudes were plotted against the original event durations (w_0) as shown in Figure 4.17. It can be seen from Figure 4.17 A that the original event duration is faithfully reproduced in the final event duration for events longer than 7 frames (illustrated by the dashed line), but after this point w_t underestimates w_0 . In analogy to a Gaussian filter, the rise time (T_r) of this particular CK filter can be considered as 7 frames or data points. Unlike a Gaussian filter, the full event amplitude is recoverable from all events lasting longer than the rise time of 7 frames, as seen from Figure 4.17 B. This is due to the ability of the CK filter to preserve sharp edges in data, at least for events longer than T_r . The relationship between w_0 and w_t for events shorter than the rise time can be approximated by a quadratic curve, enabling corrections to be made for short event lengths within real data. This quadratic fit also reveals the filter dead time, $T_d = (0.6 \pm 0.8)$ frames. Events with original durations shorter than 1 frame are only detected about 50 % of the time in the filtered intensity trace. This value for the filter dead time is a useful guide when imposing a resolution limit on event durations.

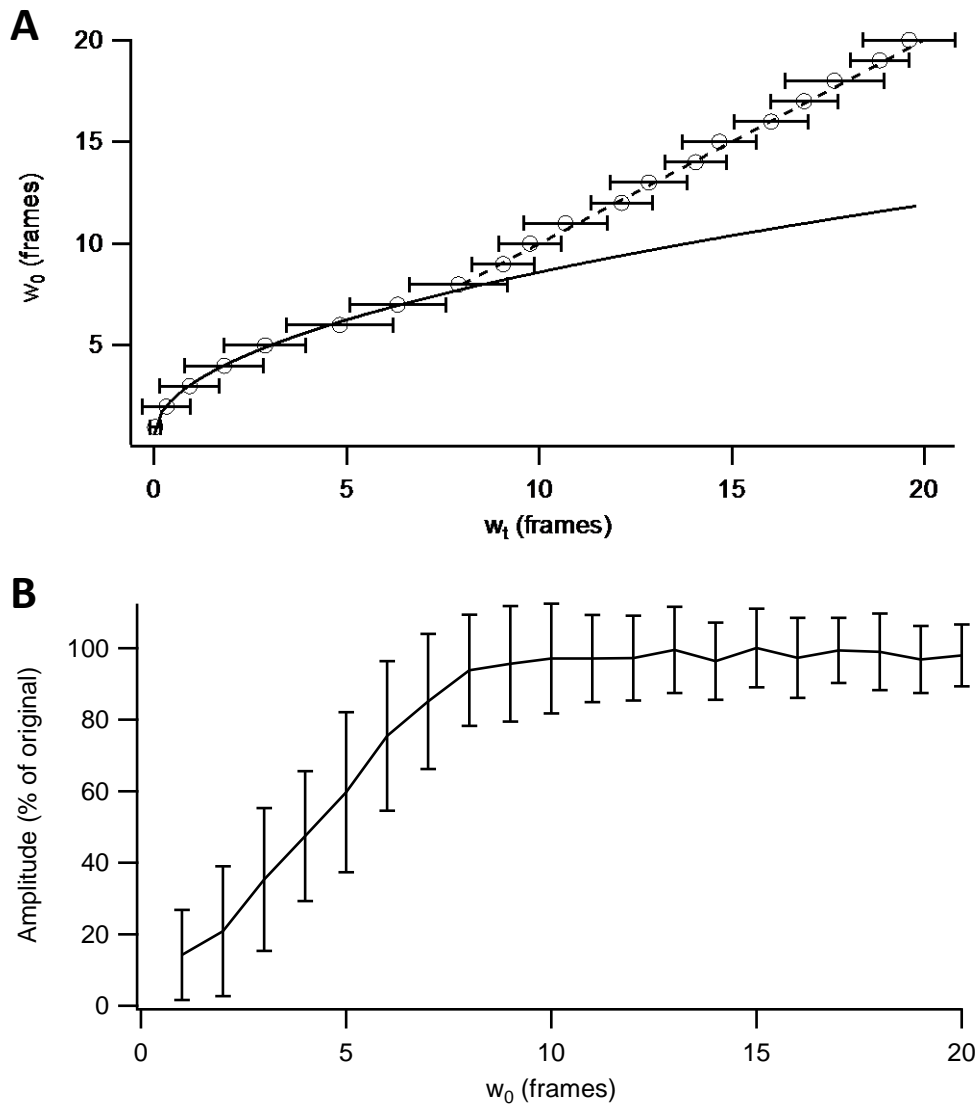


Figure 4.17. Response of Chung Kennedy filter to short duration events.

Simulated noisy data with a signal to noise ratio of 3, containing single unit-amplitude on events varying in length between 1 data point and 20 data points, were filtered using a CK filter with a bank of 4 predictors with lengths of 2, 4, 8, and 16 data points all treated as equally probable; an analysis window of 5 data points; and a p weighting factor of 50. (A) Filtered data were subjected to half-amplitude threshold analysis to determine the apparent event duration w_t , shown here against the original event duration w_0 . Error bars show the standard deviation of 50 samples. The solid line indicates a fit to the data for event durations less than 7 frames. The dashed line indicates the 1:1 relationship between w_0 and w_t for event durations longer than 7 frames. (B) The amplitudes of the on events were calculated from the filtered data as the average across the duration of the original on event w_0 . Error bars show the standard deviation of 50 samples.

C. Extract event durations and amplitudes

CK filtered intensity traces were processed using the function `halfampthresh_multilevel`. When supplied with the single event amplitude, average background intensity level, CK filter rise time in frames, and relationship between w_f and w_0 for short events, this function subtracts a baseline from the intensity traces, identifies on and off events via half-amplitude threshold analysis, and returns a list of event amplitudes and durations (in frames) corrected for short events (Figure 4.18).

On and off events were first roughly identified from the CK filtered intensity traces using the half-amplitude threshold above the average baseline. The intensity of regions identified as off events was extracted and used for an exponential baseline fit, which was then subtracted from the CK filtered intensity trace to give a zero baseline. Half-amplitude threshold analysis was performed on the baseline-corrected intensity traces to yield a list of on and off events. Event start and end times were calculated by interpolation. Durations of events shorter than the filter rise time were corrected using the quadratic relationship between w_f and w_0 , and adjusted with respect to preceding and following events. Amplitudes were extracted for events longer than T_r .

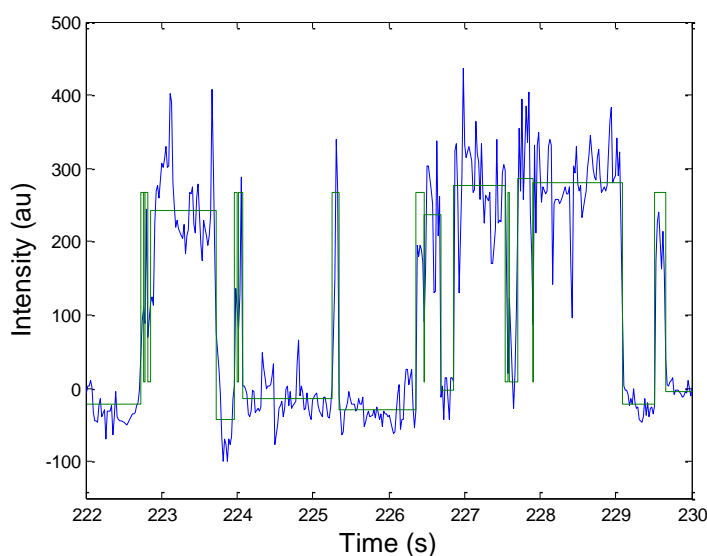


Figure 4.18. Half amplitude threshold crossing analysis enables identification of binding events.

The Chung Kennedy filtered intensity trace (blue) is overlaid with the detected binding events (green). The trace shows the change in fluorescent signal as Cy3-ATP binds to and dissociates from a ChII molecule (His₆-ChII in high salt buffer with 60 nM Cy3-ATP, viewed under 514 nm illumination with 0.01 s exposure).

D. Impose a resolution limit on event durations

As was seen in the discussion of CK filter characteristics, there is a minimum event duration which can be trusted. Events shorter than the filter dead time are not detected at all, while it is unlikely that events with durations within the error on T_d are real. It is thus necessary to impose a resolution limit on the data to exclude those events which are too short to be trusted. The resolution limit chosen here was the upper 95 % confidence limit for the filter dead time, 2.7 frames. It was imposed on the data using the function `resolutionlimit`.

Given the resolution limit in frames, the frame rate, the filter rise time and the single event amplitude, the function `resolutionlimit` removed all on and off events shorter than the resolution limit by concatenating them with preceding and following events. Off events separated by a short on event were joined to form one off event with duration equal to the three original event lengths. The same happened for on events separated by a short off event, with the new event amplitude equal to the average of the two concatenated on event amplitudes. This produced a new idealized event record with consistent time resolution (Figure 4.19).

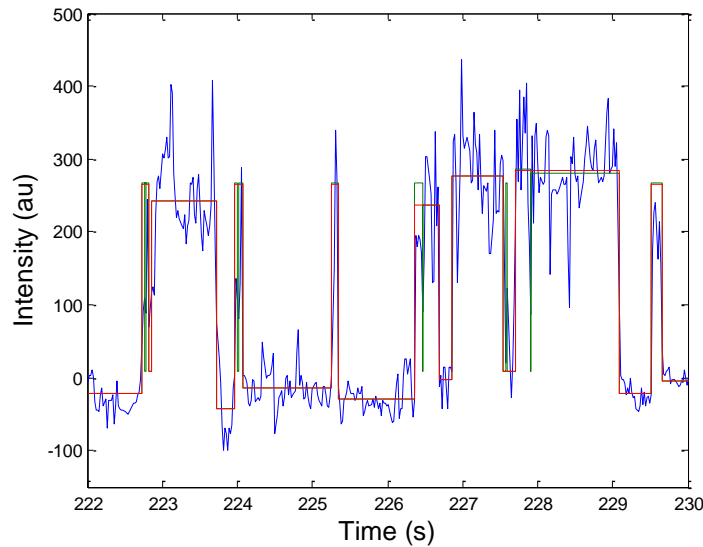


Figure 4.19. Imposing a resolution limit on event durations removes events which are shorter than the Chung Kennedy (CK) filter dead time.

The CK filtered trace (blue) is overlaid with the original detected binding events (green) and the resolution-limited binding events (red). The trace shows the change in fluorescent signal as Cy3-ATP binds to and dissociates from a ChII molecule (His₆-ChII in high salt buffer with 60 nM Cy3-ATP, viewed under 514 nm illumination with 0.01 s exposure).

4.3.3. Maximum likelihood analysis to extract time components

A

- Maximum likelihood analysis
- MLfit2

Figure 4.20. Summary of steps required for maximum likelihood analysis. The custom MATLAB functions used in each step are listed below the description.

When grouped into a histogram, binding event durations exhibit an exponential distribution (Figure 4.21). Maximum likelihood analysis tests the likelihood that a particular distribution describes the observed data by finding the parameters of the distribution which maximize the probability of observing the data. The MLfit2 function fits distributions of one and two exponentials to the on and off time data and determines statistically which is the best fit.

On and off time event durations were sorted and grouped into histograms with bin widths comparable to the inverse of the frame rate used to collect the image series. If there were any single, long duration outliers, these were excluded and the upper cut-off time of the histogram distribution was recorded. Given the resolution limit of the data (t_{res}) and the upper cut-off time if long duration outliers have been excluded, the MLfit2 function defined the negative log-likelihood functions of one- and two-component exponential distributions from the conditional probability density functions for event duration distributions given that no intervals less than t_{res} , and none longer than the upper cut-off time, are observed.

These custom negative log-likelihood functions were supplied to the MATLAB function `mle` which returned the parameter values from maximum likelihood fitting. In the case of a two-component exponential distribution, the areas associated with the components were constrained to take positive values and to sum to unity.

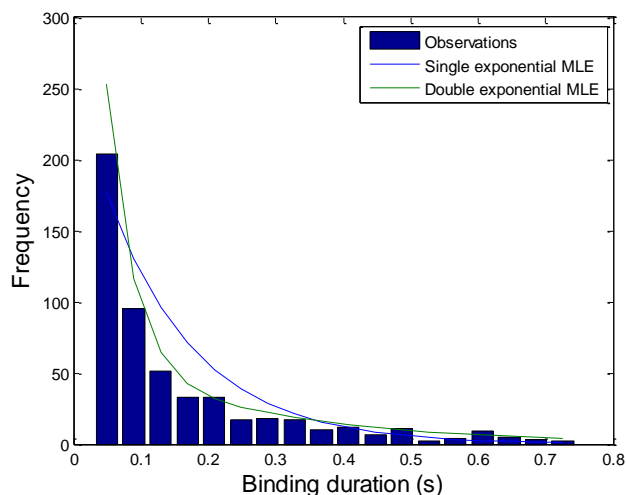


Figure 4.21. Histogram of binding event durations for a single diffraction-limited area.

Maximum likelihood estimates (MLE) for single and double exponential distributions are included. Bin width = 39.8 ms. Binding event durations are shown for Cy3-ATP binding to ChII (His₆-ChII in high salt buffer with 60 nM Cy3-ATP, viewed under 514 nm illumination with 0.01 s exposure).

4.3.4. Distinguishing between events of interest and background events

Image series from both samples (containing protein) and blanks (in the absence of protein) were subjected to MLE analysis. Unspecific binding of the fluorescent nucleotide to the functionalized surface used to immobilize the protein meant that a signal was detectable even in the blanks. Upon fitting double exponential distributions to these signals, it was possible to distinguish a subset of longer time components which was present in the sample data, but not that of the blank.

5. The stoichiometry of ChID in the ID complex

A wide range of experimental methods have been employed to probe this question, but no general consensus has as yet been reached concerning the stoichiometry and size of the active ID complex. Single particle reconstructions from cryo-EM images of the ID complex from *R. capsulatus* produced a double ring structure where 6 BchD subunits and 6 Bchl subunits formed homomeric hexamers stacked one on top of the other[14,37]. This is the only ID homologue which has been successfully isolated and subjected to high resolution structural analysis. Solution studies of the same subunits by Willows and Beale investigated the optimum ratio of I to D subunits required for maximum chelatase activity[18]. By varying the concentration of BchD in a series of stopped assays where the concentrations of Bchl and BchH were held constant, maximum activity was found for a molar ratio of 4:1 Bchl to BchD subunits, in contrast to the 1:1 stoichiometry suggested by the cryo-EM images. A single titration point at lower BchD concentration (6:1 I:D) indicated lower activity, and higher concentrations of BchD appeared to be inhibiting with a marked decrease in activity.

More rigorous titration experiments performed with recombinant subunits from *R. sphaeroides* produced different results again[19]. Varying the concentration of BchD while holding BchH fixed at saturating concentration and Bchl at limiting concentration (to investigate the interaction between I and D subunits) yielded an optimum ratio of 1:1 Bchl subunits to BchD subunits. Higher molar ratios of BchD became inhibiting. But in another titration where the concentration of H was saturating, D was limiting and I was varied, the optimum ratio was 5:1 Bchl:BchD, clearly surpassing the 1:1 molar ratio found previously. A molar ratio of 4:1:36 Bchl:BchD:BchH was settled on following a titration experiment for H where I and D were held constant and in a ratio of 4:1 I:D. It was concluded from the varying optimum ratios that titration experiments could not be used to determine the stoichiometry of the complex *in vivo*.

The results from solution experiments with *Synechocystis* subunits are equally ambiguous. Jensen et al performed a series of titration experiments varying the subunit present in saturating and limiting concentration[17]. With saturating ChIH and limiting ChID, varying the concentration of ChII produced saturation of the activity level at a ratio of 200:1 ChII to ChID subunits, a situation unlikely to be encountered *in vivo*. With ChII at limiting concentration an optimal rate was achieved at a ratio of approximately 2:1 I:D when the concentration of ChID was varied. Higher concentrations of ChID produced an inhibitory effect, similar to the findings for *R. sphaeroides*. When I and D (in the ratio 2:1) were added to limiting H, saturation was reached at a ratio of I:D:H of 2:1:4. Attempts were also made to isolate the ID complex via size exclusion chromatography and analytical ultracentrifugation[25]. Fluorescently labelled ChII subunits were mixed with ChID in the ratio 1.8:1 I:D and in the presence of nucleotide. The resulting SEC UV trace contained a peak corresponding to a molecular mass of 226 kDa, that expected for a ChII hexamer. Light scattering and SDS-PAGE performed on the 226 kDa peak fractions confirmed the presence of labelled ChII and absence of ChID which eluted in a second peak with molecular mass corresponding to ChID monomer. Analytical ultracentrifugation of equimolar quantities of labelled ChII and unlabelled ChID in the presence of ADP produced a single, well-defined

peak of approximately 120 kDa in mass. This differed from the wide spread of different species observed when labelled ChII was assayed in the absence of ChID. The mass of the single peak could correspond to a trimer of ChII subunits, or possibly a heterodimer of ChII and ChID.

Single molecule subunit counting is another, complementary method with which to address the problem. By labelling the subunits of the ID complex with fluorescent molecules, it is possible to deduce the numbers of subunits within immobilized complexes from the fluorescence intensity of the labels. In this particular instance, the number of labels was determined by monitoring the stepwise reduction of fluorescence intensity with time as fluorescent molecules photobleached under continuous laser illumination. Unlike titration experiments which can only access the overall stoichiometry of the subunits, this technique monitors the numbers of subunits within individual complexes directly.

5.1. Labelling subunits through their cysteine residues

The reactivity of the thiol side chain of cysteine makes this an attractive target for site-specific protein labelling. The amino acid sequences of ChII and ChID are shown in Figure 5.1 where the cysteine residues are shaded in red. ChII contains 4 naturally occurring cysteines while ChID contains 3. Previous work on the role of cysteine residues within *Synechocystis* chelatase subunits has revealed that they are required in the ChII subunit for ATPase activity and for chelatase activity while those in the ChID subunit are not essential for chelatase activity[32]. Specifically, it was found that incubation with *N*-ethyl maleimide (NEM, an alkylating reagent that reacts specifically with the sulfhydryl groups of cysteine residues under neutral pH conditions to permanently block these sites[115]) caused a loss of ATPase and chelatase activity in ChII. NEM-treated ChII was still able to bind to ChID, suggesting that ATP binding was not affected. By contrast, ChID was almost unaffected by treatment with NEM: chelatase activity was reported at 85 % of normal when ChID was incubated with 4 times the concentration of NEM used to inactivate ChII.

The same study found that of the 4 cysteine residues occurring in ChII, 3 are accessible to modifying agents (assayed by titration with 2,2'-dipyridyl disulphide (2-PDS), a reversible thiol-specific blocking agent which produces a chromophoric leaving group on forming pyridyl disulphide bonds, enabling the extent of thiol modification to be followed spectrophotometrically at 343 nm)[115]. A further titration assay with sub-stoichiometric ratios of NEM to protein thiol revealed that of the 3 accessible cysteines, one per ChII molecule was essential for ATPase activity. Pre-incubation of ChII with ATP, ADP, MgATP²⁻ or MgADP⁻ before modification with NEM resulted in complete protection and retention of ATPase and chelatase activity.

Since this previous work demonstrated that it was possible to modify the cysteine residues in ChII and ChID subunits in a manner which protected their activity, maleimide modification was chosen as a simple and effective method to attach fluorescent labels to them. Figure 5.2 shows the alkylation reaction between a sulfhydryl group on a protein and the double bond of a maleimide group on a fluorescent dye resulting in conjugation of the two molecules via a thioether bond[115]. Care must be taken to perform the reaction in the pH range 6.5 to 7.5 to maximize specificity for thiols.


```

sp|P51634|CHLI_SYNY3/1-357      1 MTATLAAPSKTRRVVFPFTAIVGQDEMKLALLLNVIDPKI 40
sp|P72772|CHLD_SYNY3/1-676    1 - - - MTTLTPFIPLNFIPTAIVGQEAIKLALLLGAIDPGL 36

sp|P51634|CHLI_SYNY3/1-357    41 GGVMIMGDRGTGKSTTIRALADLLPEIEVVANDPFNSSPS 80
sp|P72772|CHLD_SYNY3/1-676   37 GGIVIAGRRGTAKSVMARAHTLLPPIEIIKGNRYQCDPK 78

sp|P51634|CHLI_SYNY3/1-357    81 DPEMMSEEVRI RVD SQ - - EPLSIVKKKVTMVDLPLGATED 118
sp|P72772|CHLD_SYNY3/1-676   77 NPGSWDDDTLEKFA DVPLDQL ETQVIPA PFIQIPLGV TED 116

sp|P51634|CHLI_SYNY3/1-357   119 RVCGTIDIEKALSEG VKAFEPGLLAKANRGILYVDEVNLL 158
sp|P72772|CHLD_SYNY3/1-676  117 RLLGSVDVEKSVKQGEAVFQPGLLAQAHRGVLYIDELNLL 156

sp|P51634|CHLI_SYNY3/1-357   159 DDHLVDVLLD SAAGGWN TVEREGISIRHPARFVLVGS GNP 198
sp|P72772|CHLD_SYNY3/1-676  157 DDQIANQLLTVL TEGKNQIEREGMSFQHPQPLLIATYNP 196

sp|P51634|CHLI_SYNY3/1-357   199 EEGELRPQLLDRFGMHAEIRTVREPELRVKIVEQRTEFDQ 238
sp|P72772|CHLD_SYNY3/1-676  197 EEGPLRRHLLDRIAIALSADGILGLDQRVAAVDQVLAYAD 236

sp|P51634|CHLI_SYNY3/1-357   239 NHPFCDDQYQTEQEALQAKIVNAQNLLPQVTIDYDYRVKV 278
sp|P72772|CHLD_SYNY3/1-676  237 SPISFIDQYDAELDDLKTTIILAREWLKEVSLTPEQVSYL 276

sp|P51634|CHLI_SYNY3/1-357   279 SEVCAELDV DGLRGDIVTNRAAKALAAFEGRTEVTVD DIS 318
sp|P72772|CHLD_SYNY3/1-676  277 VEEAIRGGLQGHRGELFAMRVAKAIAALDGRSDVQADDLR 316

sp|P51634|CHLI_SYNY3/1-357   319 RVIVLC LRHR - - - - - 328
sp|P72772|CHLD_SYNY3/1-676  317 QAVELVIVPRSVLMDNPPPPPEQAPPPPPPPQNQDEGKDEQ 356

sp|P51634|CHLI_SYNY3/1-357   329 - - - - - LRKDPLESIDSGSKVEKVFKRVFVVDEA 357
sp|P72772|CHLD_SYNY3/1-676  357 EDQQDDKEDDKDNEPEAEQDPPSIP EEFIFDPEGVSLDPS 396

sp|P51634|CHLI_SYNY3/1-357   - - - - -
sp|P72772|CHLD_SYNY3/1-676  546 ETLP CGGGSPLIADM 675

```

Figure 5.1. Amino acid sequence alignment of ChII and ChID.

Conserved residues are shaded in blue. The regions picked out with coloured borders are the Walker A motif (yellow), the Walker B motif (cyan), the arginine finger (pink) and sensor 2 (dark blue). Cysteine residues are shaded red. The C terminal regions of the ChID sequence which extend beyond the sequence of ChII and which do not contain a cysteine residue have been hidden (indicated by a vertical blue line). Sequences were obtained from the UniProt Knowledgebase[102] and aligned in Jalview[135] using Clustal W[136].

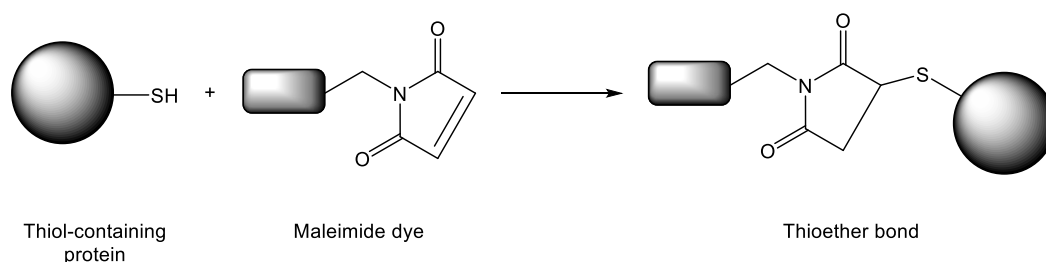


Figure 5.2. Maleimide modification of a thiol group to form a thioether bond.

5.1.1. Determining labelling success and efficiency

ChII and ChID subunits were labelled through their cysteine residues with monofunctional maleimide fluorescent dyes. After removing excess unreacted dye from the protein solution by gel filtration, the success of the labelling reaction was indicated by the intense colour of the protein solution and verified by SDS-polyacrylamide gel electrophoresis (SDS-PAGE). Before the gel was developed for protein visualization the dye fluorescence could be observed by illuminating the gel with UV or visible light of the appropriate wavelength. Figure 5.3 displays examples of gels of fluorescently labelled protein, with labelled protein shown in panels B, D and F, and the corresponding total protein shown in panels A, C and E.

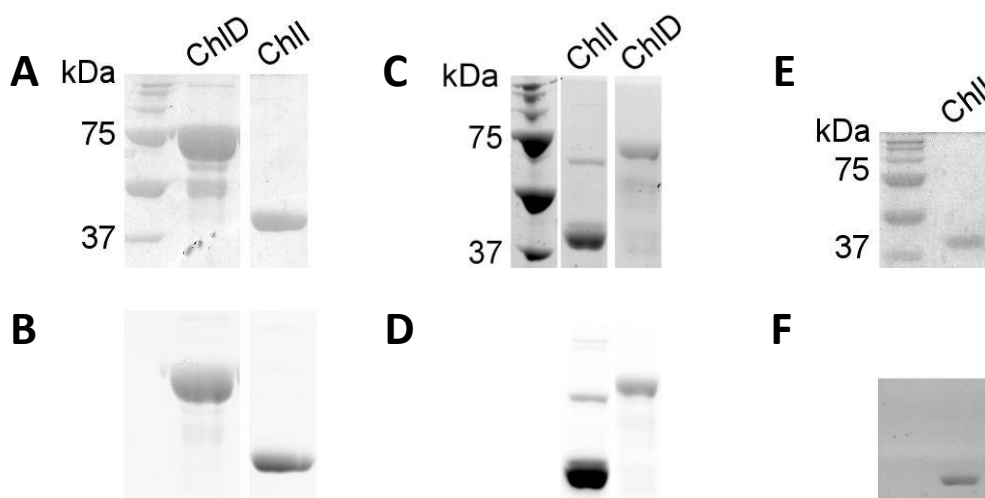


Figure 5.3. SDS-PAGE gels demonstrating ChII and ChID subunits labelled with fluorescent dyes.

(A) Subunits labelled with Cy3-maleimide visualized by Coomassie staining and (B) under UV illumination revealing the fluorescent dye. (C) Subunits labelled with Cy5-maleimide in a no-stain gel visualized after 5 min activation with UV illumination and (D) under 647 nm illumination revealing the fluorescent dye. (E) ChII labelled with Alexa Fluor 532 visualized by Coomassie staining and (F) under UV illumination revealing the fluorescent dye. The greyscales in images B, D and F have been inverted to improve visibility.

The labelling efficiency of a reaction was determined as the ratio of the fluorescent dye concentration to the protein concentration of a solution (Equation 2.3). For ChID subunits and for unprotected labelling (labelling in the absence of MgATP^{2-}) of ChII subunits the concentrations of dye and protein were determined spectrophotometrically, applying the appropriate correction factor for the absorption of the dye at 280 nm. For ChII subunits labelled in the presence of MgATP^{2-} the protein concentration was determined using the

Bradford assay (see Section 2.8.2.). Table 5.1 summarizes the mean labelling efficiencies achieved when labelling ChII and ChID subunits with various fluorescent dyes.

Table 5.1. Mean labelling efficiencies for ChII and ChID subunits labelled with fluorescent dyes.

Singly-labelled ChII was obtained by including MgATP^{2-} in the thiol modification buffer to protect 3 of its 4 cysteine residues (protected ChII).

Protein	Label	Mean labelling efficiency (dye molecules per protein molecule)	Standard deviation (dye molecules per protein molecule)
Unprotected ChII	Alexa Fluor 532	1.8	0.9
	Cy3	1.07	0.04
Protected ChII	Cy3	0.85	0.07
ChID	Cy3	1.17	0.03
	Cy5	1.4	0.3

When a dye was allowed to react with any of the available cysteines in ChII (unprotected labelling), the resulting labelling efficiency was variable. Both Cy3 and its Alexa Fluor 532 equivalent were used. On average Alexa Fluor 532 produced protein molecules labelled with 2 dye molecules, although the range of labelling efficiencies (0.5 to 2.8 dye molecules per protein molecule) indicates that it is possible to modify between 1 and 3 of the accessible cysteines in a given reaction. The labelling efficiency was difficult to control reliably when reacting unprotected ChII with Alexa Fluor 532. Repeats of the labelling reaction produced labelling efficiencies ranging from 1 to 3 dye molecules per protein. By contrast, unprotected labelling of ChII with Cy3 resulted in a labelling efficiency with a much smaller spread about the mean of 1.07 dye molecules per protein molecule. Although 3 cysteines are accessible for modification, unprotected labelling with Cy3 tends to produce singly labelled ChII subunits. Protected labelling of ChII resulted in a similarly consistent labelling efficiency of around 1 dye molecule per protein molecule.

The mean labelling efficiencies greater than 1 dye molecule per protein molecule attained for modification of ChID suggest that of the 3 cysteine residues present in ChID, 2 are more easily accessible. Given that the greatest labelling efficiency obtained was 1.6 Cy5 molecules per ChID molecule, it is likely that 2 of the 3 cysteine residues in ChID are accessible to modifying agents. On average, labelling ChID with Cy3 produced protein molecules carrying 1 dye molecule while labelling with Cy5 produced protein molecules carrying 1 or 2 dye molecules. A single labelling reaction using Alexa Fluor 532 produced a labelling efficiency of 1.4 dye molecules per ChID molecule.

5.1.2. Activity of labelled subunits

Magnesium chelatase assays were performed to determine if labelled subunits retained their activity. Figure 5.4 shows examples of calibration and progress curves from assays carried out on unlabelled and labelled subunits. Assays were always run with 0.1 μM ChII (unlabelled or labelled), 0.1 μM ChID (unlabelled or labelled) and 0.4 μM ChIH (unlabelled). Calibration curves were constructed from magnesium deuteroporphyrin IX (MgD_{IX}) standards of known concentration in chelatase buffer containing magnesium chelatase subunits. Calibration curves were also constructed from standards in the presence of

fluorescently labelled subunits to compensate for any interference which the fluorescent dye might cause in detecting the fluorescent signal from MgD_{IX} (excitation/emission at 420/580 nm). It can be seen from the example in Figure 5.4 A that dyes with excitation/emission wavelengths in the region of 532/550 nm (Alexa Fluor 532 and Cy3) do not affect the fluorescence detection of MgD_{IX} .

A typical chelatase progress curve with unlabelled subunits (Figure 5.4 B, solid line) shows a lag time of approximately 10 min before MgD_{IX} production reaches a steady rate. The concentration of MgD_{IX} plateaus after around 50 min. When labelled ChID subunits are substituted in, the reaction proceeds more slowly and results in a slightly lower final product concentration. Labelled ChII subunits show no chelatase activity.

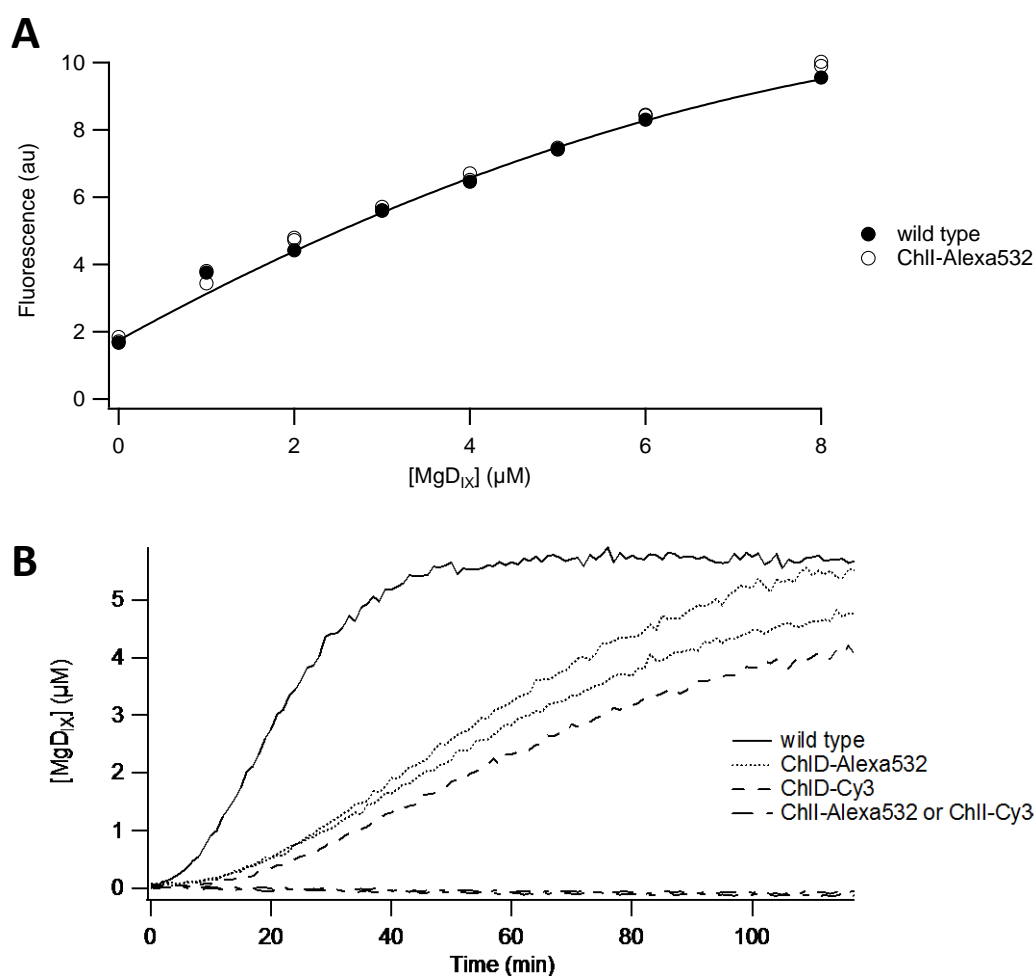


Figure 5.4. Example calibration and progress curves from a magnesium chelatase assay.

(A) Calibration curves constructed from MgD_{IX} standards in the range 0 μM to 8 μM dissolved in standard chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 34 °C) in the presence of 0.1 μM ChID, 0.4 μM ChIH, and 0.1 μM ChII (filled circles) or 0.1 μM ChII-Alexa532 (open circles). The solid line is a 2nd degree polynomial fit to the data points. (B) Progress curves of magnesium chelatase assays run in standard chelatase buffer containing 8 μM D_{IX} , 5 mM MgATP^{2-} and 10 mM free Mg^{2+} in the presence of unlabelled subunits (solid line); ChID-Alexa532 (dotted line); ChID-Cy3 (dashed line); or ChII labelled with Alexa Fluor 532 or Cy3 (dot-dashed line). Subunit concentrations were 0.1 μM ChII, 0.1 μM ChID and 0.4 μM ChIH in all assays.

Table 5.2 summarizes the mean steady state rates for chelatase assays performed with unlabelled and labelled subunits. On average, labelled ChID exhibits 30 % of the normal chelatase activity.

Table 5.2. Mean steady state rates of magnesium chelatase assays performed at saturating substrate concentrations.

His₆-ChID and His₆-ChIH subunits were used throughout and ChII subunits were untagged unless indicated. Where indicated, a fluorescently-labelled subunit was substituted for the unlabelled subunit. Assays were run with 0.1 μ M ChII, 0.1 μ M ChID and 0.4 μ M ChIH in standard chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 34 °C) containing 8 μ M D₁₉, 5 mM MgATP²⁻ and 10 mM free Mg²⁺.

Protein	Mean steady state rate (μ M \cdot min ⁻¹)	Standard deviation (μ M \cdot min ⁻¹)	Proportion of unlabelled steady state rate (%)	Standard deviation (%)
Unlabelled				
ChII	0.21	0.07	—	—
His ₆ -ChII	0.34	0.16	—	—
Labelled				
ChII-Alexa532	No activity	—	—	—
ChII-Cy3	No activity	—	—	—
ChID-Alexa532	0.070	0.009	34	4
ChID-Cy3	0.052	0.004	29	1
ChID-Cy5	0.049	0.001	28	1

The reduction or abolishment of chelatase activity observed in labelled subunits of ChID and ChII respectively disagrees with previous findings that modification of cysteines can be performed with little adverse effect on activity[32]. However that study used the relatively small modifier NEM. Fluorescent dye molecules are much larger, which means they might cause steric hindrance in the protein molecules they bind to (Figure 5.5). In another study dealing with the labelling of ChII subunits, a series of single cysteine mutants was made, replacing 1 of the 4 cysteines with serine[25]. The C121S mutant was found to be inactive, with circular dichroism (CD) spectroscopy revealing it did not fold correctly. Upon labelling the remaining 3 mutants (protected by the presence of ADP) with the fluorescent dye fluorescein-5-maleimide, 2 were found to retain some chelatase activity (C234S and C244S). This supports the theory of steric hindrance by dye molecules, in that different combinations of available cysteines produce varying effects on the activity when the subunit is fluorescently labelled.

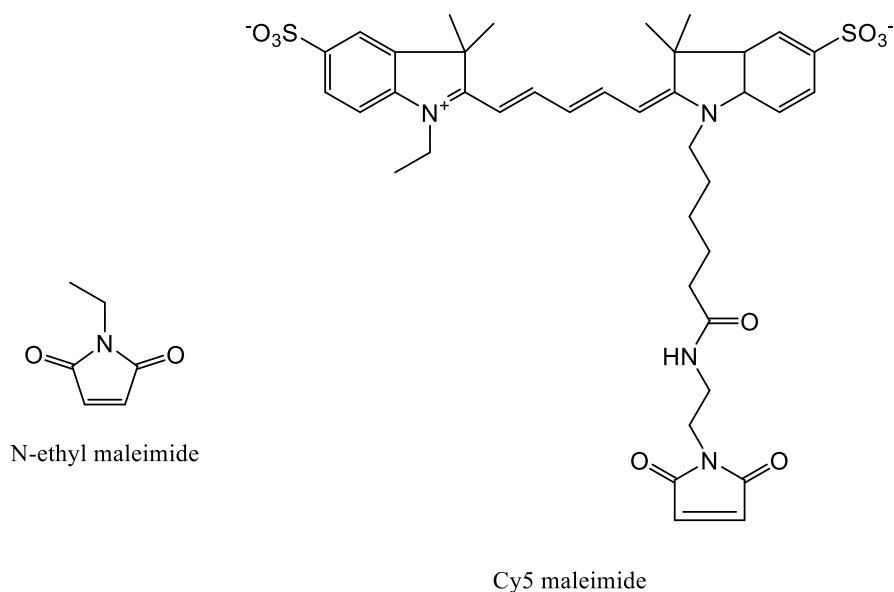


Figure 5.5. Chemical structures of thiol-reactive modifiers N-ethyl maleimide and Cy5 maleimide.

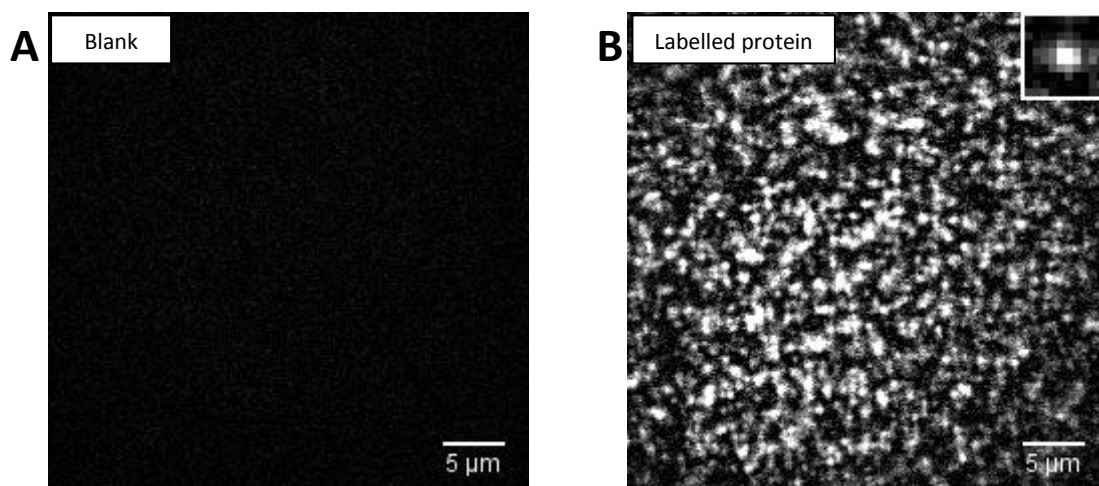


Figure 5.6. Fluorescently-labelled ChII binding to ChID.

Subunits were incubated together at a concentration of 6 μM each in standard chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 25 $^{\circ}\text{C}$) with 5 mM MgATP^{2-} and 10 mM free Mg^{2+} for 5 min before rapid dilution to 1 nM concentration and application to a Ni-NTA functionalized coverslip. (A) Ni-NTA surface in thiol-free chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 $^{\circ}\text{C}$) with 5 mM MgATP^{2-} and 10 mM free Mg^{2+} viewed under 532 nm laser illumination. (B) Ni-NTA surface with 1 nM ChII protectively labelled with Alexa Fluor 532 and 1 nM $\text{His}_6\text{-ChID}$ under the same conditions as for (A). An image of a single fluorophore is shown enlarged $\times 4$ in the top right corner.

While labelled ChII subunits lose their chelatase activity, they are still capable of binding to ChID. Figure 5.6 shows ChI subunits protectively labelled with Alexa Fluor 532 (ChII-Alexa532) in the presence of MgATP^{2-} and $\text{His}_6\text{-ChID}$. The $\text{His}_6\text{-ChID}$ is immobilized on a Ni-NTA functionalized coverslip, and ChII-Alexa532 molecules binding to the immobilized ChID molecules are visible under 532 nm laser illumination. Pull down assays using ChII labelled with fluorescein-5-maleimide also demonstrate that labelled ChII is still capable of binding

to ChID[25]. This suggests that fluorescently labelled ChII can bind the MgATP^{2-} necessary to form a complex with ChID but cannot hydrolyse it, as was found for unprotected, NEM-treated ChII[32]. It is unclear whether this is a full catalytic complex given the lack of chelataase activity.

Given that fluorescently labelled ChID retained some chelataase activity, while producing active and labelled ChII required more modifications to be made to the protein, it was decided to use labelled ChID in single molecule stoichiometry studies. This subunit already carries a His_6 tag due to the difficulty in purifying ChID in sufficient quantities by any method other than nickel affinity chromatography. The His_6 tag provided a ready and simple means of immobilizing the protein on a coverslip. Immobilizing the labelled subunit in a complex also reduces the possibility of undercounting due to dissociation of subunits during imaging at the low concentrations required for single molecule microscopy.

5.2. Determining optimum protein concentration on a coverslip

Varying protein concentration during surface immobilization has a substantial effect on the final density of fluorescent molecules (Figure 5.7). 20 pM to 60 pM gives a good coverage without being too dense. Varying the volume of buffer used to wash away unbound protein after the incubation does not have much of an effect, at least for volumes over 200 μl (Figure 5.8). This shows the protein is reasonably stably immobilized. Comparison of the density of immobilized fluorophores in Figure 5.7 and Figure 5.8 also highlights the effect of a longer incubation time on the coverslip. The density of immobilized fluorophores evident in protein incubated at 60 pM for 1 min on a coverslip (Figure 5.7) is comparable to that of protein incubated at 25 pM for 5 min (Figure 5.8). In single molecule stoichiometry experiments, a concentration of 60 pM was chosen with an incubation period of 1 min.

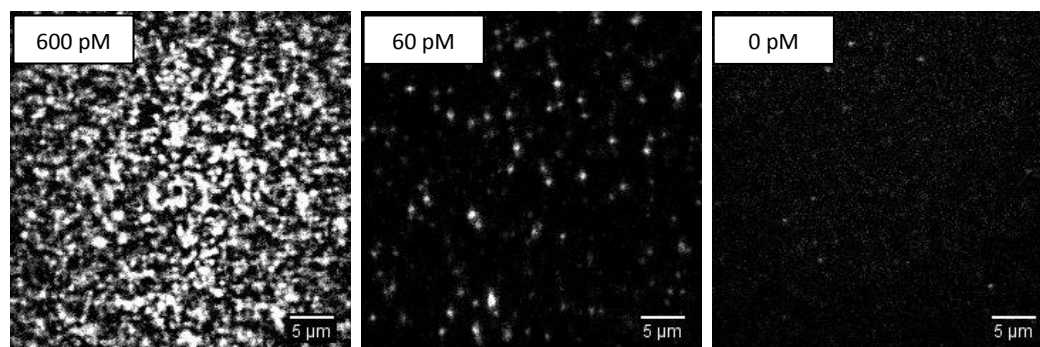


Figure 5.7. Effect of varying protein concentration on the density of fluorescent molecules visible on a coverslip.

6 μM ChII was incubated with 6 μM His_6 -ChID-Alexa532 in standard chelataase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 25 $^{\circ}\text{C}$) with 5 mM MgATP^{2-} and 10 mM free Mg^{2+} for 5 min before rapid dilution into thiol-free chelataase buffer and application to a Ni-NTA functionalized coverslip. Unbound protein was washed off after an incubation period of 1 min on the coverslip. Fluorescently-labelled protein was visualized under 532 nm laser illumination in TIR mode at the picomolar concentrations indicated in the top left corners of the images.

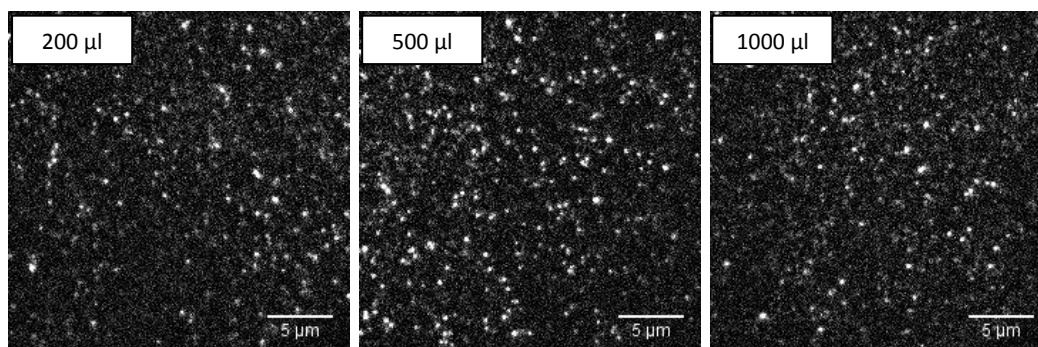


Figure 5.8. Effect of varying wash buffer volume on the density of fluorescent molecules visible on a coverslip.

1 μM ChII was incubated with 1 μM His₆-ChID-Cy5 in standard chelataase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 25 °C) with 5 mM MgATP²⁻ and 10 mM free Mg²⁺ for 5 min before rapid dilution into thiol-free chelataase buffer and application to a Ni-NTA functionalized coverslip at a concentration of 25 pM. After a 5 min incubation on the coverslip, unbound protein was washed away with varying volumes of chelataase buffer (indicated in the top left corners of the images). His₆-ChID-Cy5 was visualized under 647 nm laser illumination in TIR mode.

5.3. Distinguishing between labelled subunits and background fluorescence

Using high quality glass substrates with very high internal glass quality (very few bubbles, streaks, etc.) is essential for single molecule microscopy. However, glass substrates generally exhibit some intrinsic fluorescence or scattering due to impurities, and modification of the surface for functionalization can introduce impurities. The quality of the images produced and therefore the ability to distinguish between the fluorophores of interest and background fluorescence depends on the quality of the functionalized surface. Consider the 0 pM frame from Figure 5.7. The impurities visible on this surface could be mistaken for fluorescent dye molecules without comparison to the 60 pM frame, where the fluorophores are evidently brighter.

The level of background fluorescence varied between samples, depending on the quality of the substrate. Figure 5.9 illustrates the background intensity varying between 75 au and 250 au for different samples under the same illumination conditions. It is still possible to distinguish fluorophores against high background intensities in these examples because the fluorescent signal is at least twice as large as the noise, although in the 2500 pM sample the close spacing of the fluorophores makes it more difficult to distinguish individual signals.

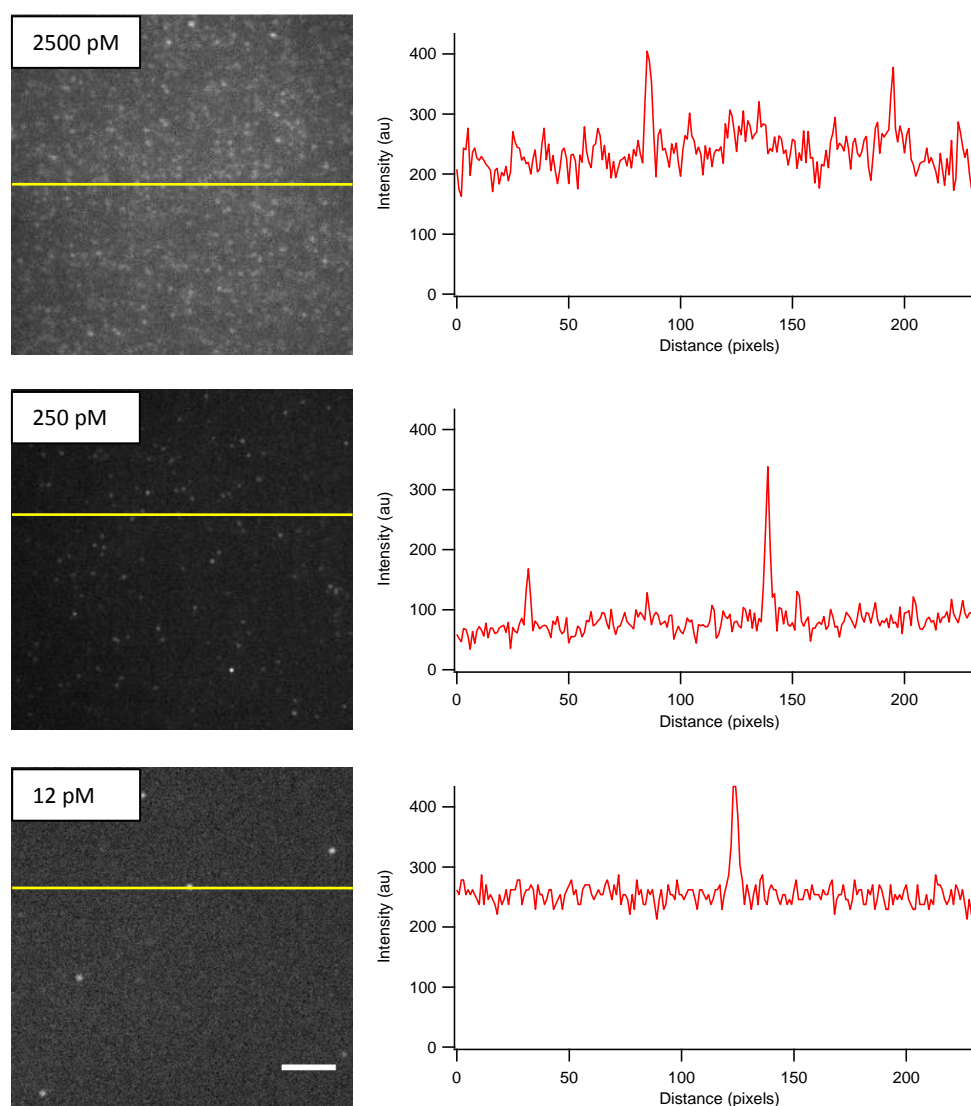


Figure 5.9. The fluorescence emission from immobilized fluorophores is visible above background fluorescence intrinsic to the sample coverslip.

Images of Cy3-labelled ChID immobilized at varying concentration (indicated in the top left of each frame). Bar = 5 μm . Pixel intensities from the linescans highlighted in yellow are displayed next to each image. Labelled protein was incubated for 1 min on the coverslip before rinsing and viewed in imaging buffer (50 mM MOPS/KOH, 0.3 M glycerol, glyoxy, pH7.7 at 25 $^{\circ}\text{C}$) under 514 nm laser illumination in TIR mode with a 100 ms exposure.

5.4. Investigating the stoichiometry of ChID in the ID complex

Two different approaches (step counting and total intensity analysis) can be used to determine subunit stoichiometries. In step counting, the intensity of a fluorescent spot is monitored over time and the number of discrete reductions in intensity (steps) is counted. The number of steps observed indicates the number of individual fluorophores present in the fluorescent spot. When a population of fluorescent spots is analyzed in this manner, the results are summarized in a histogram showing the distribution of fluorophore numbers observed in each spot. The distribution is fitted with a binomial model to determine the underlying stoichiometry. Total intensity analysis works by fitting the distribution of

intensities of the individual fluorescent spots (since the total intensity of a spot is proportional to the number of fluorophores present) to Gaussian distributions to determine the numbers of fluorophores present. The weights of the Gaussian distributions indicate the relative numbers of fluorescent spots containing a certain number of fluorophores.

The total intensity analysis approach has been employed as an alternative to photobleaching step counting in studies investigating diffusing fluorescent spots[116–118] where high frame rates and high laser intensities cause rapid photobleaching and blinking of fluorophores, making it impossible to detect single photobleaching events. While it is possible to directly determine the underlying stoichiometry by monitoring the change in total intensity distribution over time[116], this technique was not deemed necessary since photobleaching steps were easily observed in the data, and it was not possible to develop the models and code for such an analysis within the time constraints of the project.

ChII was incubated with fluorescently labelled His₆-ChID at a concentration of 3 μ M to 6 μ M each for 5 min in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH 7.7 at 25 °C) containing 5 mM MgATP²⁻ and 10 mM free Mg²⁺ before rapid serial dilution in the same buffer (minus DTT) to a concentration of 60 pM. Subunits were then immobilized on a Ni-NTA functionalized coverslip and the fluorescence from the labels on the ChID subunits was recorded under 532 nm or 647 nm laser illumination (depending on the label) using an inverted fluorescence microscope. The resulting image series were subjected to analysis as detailed in Sections 4.1. and 4.2. to identify fluorescent spots (labelled ChID subunits in a complex with ChII), extract their fluorescence intensity with time, and determine the number of photobleaching events (abrupt drops in the intensity level) occurring in each intensity trace. Figure 5.10 presents examples of photobleaching traces and fluorophore images.

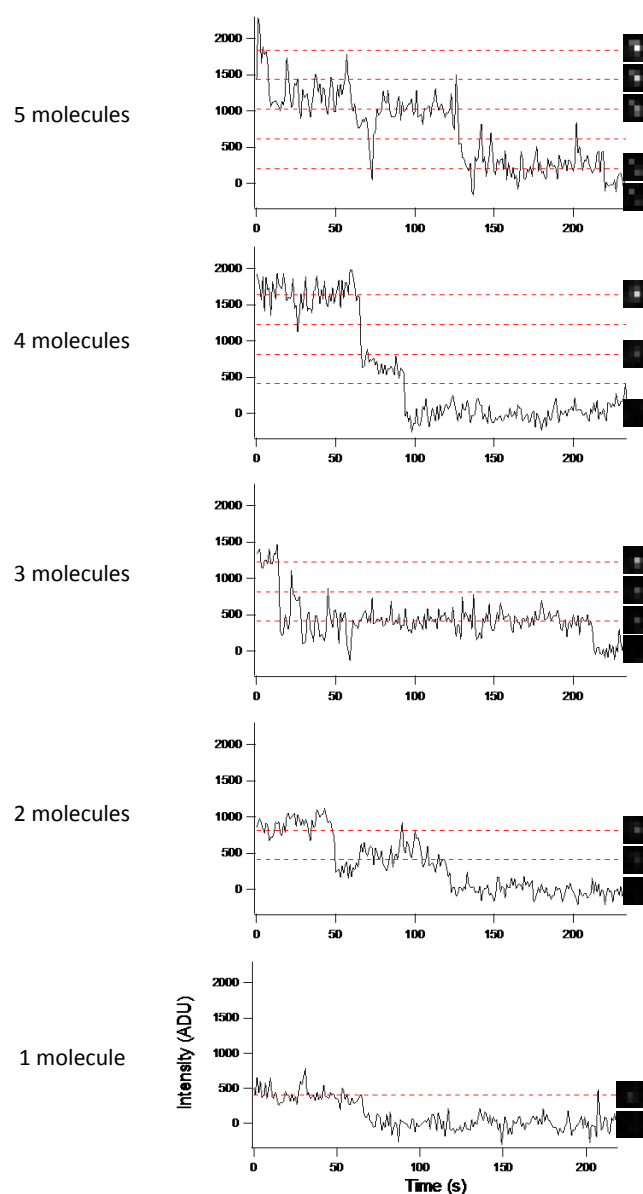


Figure 5.10. Example photobleaching intensity traces.

Example photobleaching traces for Cy3-labelled His₆-ChlD in complex with ChII, viewed under 532 nm illumination with 1 s exposure. Dashed red lines indicate intensity levels corresponding to multiples of the mean single molecule intensity determined from single Cy3 molecules (Section 4.2.1. C). Images of the fluorophores (1.3 μm \times 1.3 μm) are shown where sufficient time has been spent at a given intensity level.

The results are displayed in histograms showing the number of complexes observed to possess a certain number of labels, such as those in Figure 5.11. Previous single molecule subunit counting experiments have shown that these distributions exhibit a range of label numbers per complex rather than a single peak at the expected stoichiometry[43,72,73,79]. In fact, the observed label number (k) follows a binomial distribution governed by θ , the probability that the label on a particular subunit will be detected, and n , the actual number of subunits in a complex. When this probability distribution is multiplied by the total number of complexes observed (N_t), it gives the expected label number distribution:

$$N(k) = N_t \binom{n}{k} \theta^k (1 - \theta)^{n-k} \quad \text{where} \quad \binom{n}{k} = \frac{n!}{(n-k)!k!}$$

Equation 5.1. Binomial distribution for singly labelled subunits assembled into a complex of n subunits

The value of θ depends not only on the labelling efficiency of the subunits, but on experimental factors affecting the observed fluorescence such as fluorescence quenching, pre-experiment photobleaching, or correct folding and maturation specifically in the case of fluorescent proteins, and the ability to detect step-like events such as poor signal to noise ratios (SNRs)[72,73].

Equation 5.1 is valid in cases where subunits are labelled with up to 1 fluorescent label each. If the labelling efficiency exceeds this level, as has been shown to happen when labelling ChID with Cy5, each observed label cannot be interpreted as representing a different subunit. The simple binomial distribution must be adjusted to account for the fact that each subunit might now be labelled with a second dye molecule with probability β . The number of labels present in a complex consisting of n subunits can now range between n and 2n. If a_i is defined as the probability that an n-subunit complex contains i subunits with 2 labels (itself following a binomial distribution), the observed label number distribution can be written as a sum of binomial distributions weighted by the probability of a complex occurring with (n + i) labels:

$$N(k) = N_t \cdot \sum_{i=0}^n a_i \binom{n+i}{k} \theta^k (1 - \theta)^{n+i-k} \quad \text{where} \quad a_i = \binom{n}{i} \beta^i (1 - \beta)^{n-i}$$

Equation 5.2. Distribution for multiply labelled subunits assembled into a complex of n subunits

In this case θ represents the probability that a particular fluorescent label will be detected.

Figure 5.11 shows the distributions of observed number of labels in each complex for ChID tagged with Cy5 at labelling efficiencies of 1.5 dye molecules per subunit (A) and 1.3 dye molecules per subunit (B). The lines represent fits to the data using Equation 5.2 with β equal to 0.5 and 0.3 for labelling efficiencies of 1.5 and 1.3 dye molecules per subunit respectively (assuming every subunit carries at least one label). The results from fitting the distributions are given in Table 5.3. The data in A appear reasonably well described by the fits assuming 3- or 4-subunit complexes, but are equally well described by fits to the simple binomial distribution in Equation 5.1 assuming 6- or 7-subunit complexes (shown in Figure 5.11C). The distribution in B is equally poorly described by Equation 5.1 and Equation 5.2.

This highlights the problem that when the probability of observing a particular fluorescent label is relatively low (the maximum value found for θ was 0.6, for the 3-subunit multiple label fit to the data in A), it is difficult to determine a unique and optimal estimate for the parameters with sufficient confidence[83]. The problem is exacerbated when it is possible to observe multiple photobleaching steps associated with a single subunit. Therefore, the rest of the data concentrate on Cy3-labelled ChID for which labelling efficiencies of 1 dye molecule per subunit were obtained.

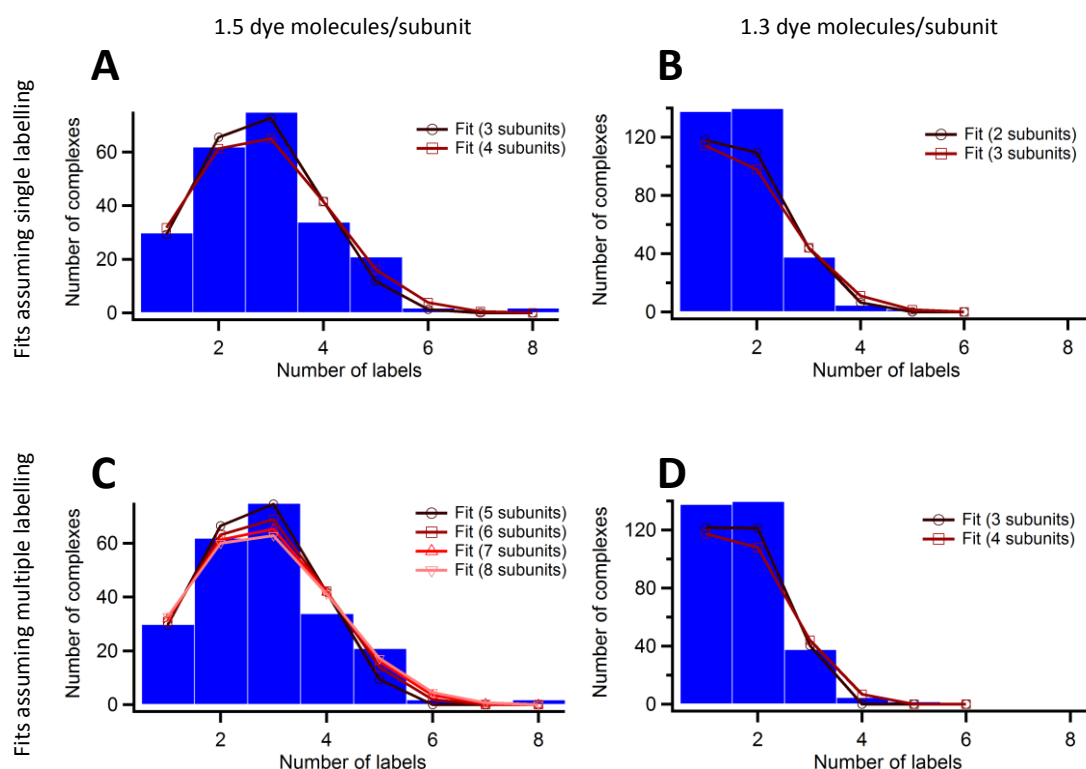


Figure 5.11. Histograms of single molecule photobleaching steps for ChlD multiply labelled with Cy5 in complex with ChlI.

Histograms are shown for ChlD with a mean labelling efficiency of 1.5 dye molecules per subunit (left hand column) and 1.3 dye molecules per subunit (right hand column), overlaid with fits to binomial distributions for varying numbers of ChlD subunits per complex (indicated in figure legends). The top row displays fits assuming ChlD subunits are labelled with 1 or 2 dye molecules each; the bottom row displays fits assuming a labelling efficiency of 1 dye molecule per subunit.

Table 5.3. Fitting parameters for the binomial distributions used to describe the histograms of single molecule photobleaching steps shown in Figure 5.11. n: number of subunits in a complex, θ : probability of detecting a fluorescent step for a particular fluorescent label.

Labelling efficiency (dye molecules per protein molecule)	n	θ	Mean square weighted deviation (χ^2_{red})
Fit to Equation 5.2			
1.5	3	0.60 ± 0.02	5.8
1.5	4	0.46 ± 0.02	5.3
Fit to Equation 5.1			
1.5	5	0.52 ± 0.02	8.2
1.5	6	0.45 ± 0.01	5.2
1.5	7	0.39 ± 0.01	5.3
1.5	8	0.34 ± 0.01	6.3
Fit to Equation 5.2			
1.3	2	0.45 ± 0.04	30.9
1.3	3	0.30 ± 0.05	55.3
Fit to Equation 5.1			
1.3	3	0.50 ± 0.03	17.0
1.3	4	0.38 ± 0.04	33.6

Figure 5.12 shows the distribution of fluorophores observed in ID complexes for ChID labelled with Cy3 at an efficiency of 1 dye molecule per subunit. Panel A displays fits to the data using Equation 5.1, assuming complexes comprising 4 and 5 subunits. In this data set, images were recorded at 4x binning rather than 1x binning to improve the SNR. (Binning relates to how the pixels of the camera are read out. 4x binning means that every pixel in the final recorded image is comprised of 16 actual pixels, the value for the final pixel being the sum of the counts recorded in each of the 16 individual pixels. In this case, using a 100x objective meant that each pixel in the final image measured 260 nm × 260 nm.) The binning operation introduced the possibility that two complexes could occur within the same resolution-limited area. The expected distribution of label numbers in this case can be represented as the sum of two binomial distributions describing a single complex and two overlapping complexes in proportions A and 1 – A respectively:

$$N(k) = N_t \left[A \binom{n}{k} \theta^k (1 - \theta)^{n-k} + (1 - A) \binom{2n}{k} \theta^k (1 - \theta)^{2n-k} \right]$$

Equation 5.3. Sum of binomial distributions for singly labelled subunits allowing for 2 complexes of n subunits to occupy the same resolution-limited area

Panel C of Figure 5.12 displays the results of fitting Equation 5.3 to the data assuming complexes of 2 and 3 subunits. While it can be seen from the corresponding χ^2_{red} values in Table 5.4 that the fit with Equation 5.3 and n = 3 provides a marginally better match to the data, this is not uniquely the best fit.

An issue arising from 4x binning is an increase in the detection of false fluorescence signals. As well as increasing the brightness of the fluorophores of interest, 4x binning exaggerates any fluorescence signal intrinsic to the substrate. This was quantified by imaging a blank sample (prepared in the same manner as a normal sample, but omitting the protein incubation step) under the same conditions and subjecting it to the same analysis. The histograms in panels B and D of Figure 5.12 display the same label number distribution after correction for the frequency of events detected in a blank. This correction does not greatly alter the fit results, with the fit to Equation 5.3 and n = 3 appearing marginally better again.

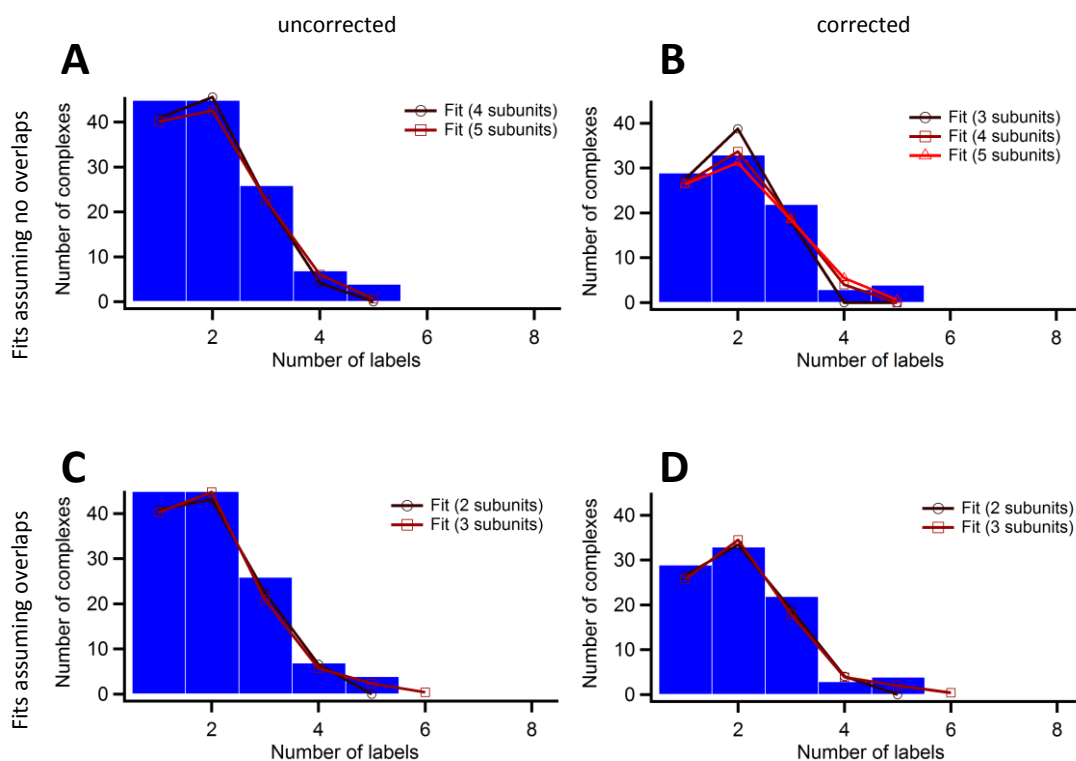


Figure 5.12. Histograms of single molecule photobleaching steps for ChID singly labelled with Cy3 in complex with ChII.

Histograms are shown for ChID with a mean labelling efficiency of 1 dye molecule per subunit as detected (left hand column), and corrected for background fluorescence events (right hand column), overlaid with fits to binomial distributions for varying numbers of ChID subunits per complex (indicated in figure legends). The top row displays fits assuming ID complexes are well separated; the bottom row displays fits allowing two complexes to be closer together than the resolution limit.

Table 5.4. Fitting parameters for the binomial distributions used to describe the histograms of single molecule photobleaching steps shown in Figure 5.12. n: number of subunits in a complex, θ : probability of detecting a fluorescent step for a particular fluorescent label, A: proportion of complexes which do not overlap.

Type of fit	n	θ	A	Mean square weighted deviation (χ^2_{red})
Raw histogram				
Well separated complexes	4	0.43 ± 0.02	-	5.3
Well separated complexes	5	0.35 ± 0.02	-	4.3
Overlapping complexes	2	0.54 ± 0.14	0.39 ± 0.35	6.8
Overlapping complexes	3	0.51 ± 0.03	0.81 ± 0.08	3.5
Corrected histogram				
Well separated complexes	3	0.59 ± 0.03	-	7.5
Well separated complexes	4	0.46 ± 0.02	-	3.9
Well separated complexes	5	0.37 ± 0.02	-	4.6
Overlapping complexes	2	0.46 ± 0.08	0.02 ± 0.32	5.9
Overlapping complexes	3	0.56 ± 0.03	0.85 ± 0.08	2.9

Correcting for false event detections introduces greater uncertainty into the final label number distribution. It was found that reducing the exposure time reduced the false event detection rate to a level which was negligible compared to the sample distribution. Figure 5.13 displays the label number distribution resulting from imaging Cy3-labelled ChID in complex with ChII with a shorter exposure time. However, the optimal fit is again ambiguous.

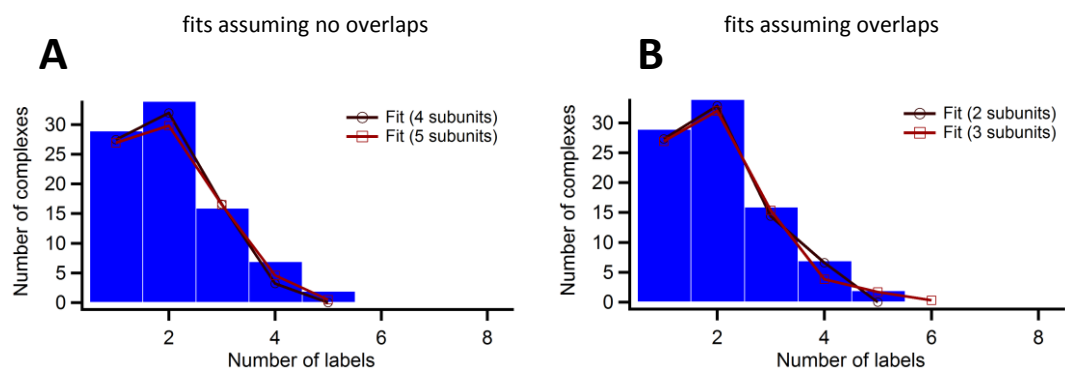


Figure 5.13. Histogram of single molecule photobleaching steps for ChID singly labelled with Cy3 in complex with ChII.

ChID was labelled with an average of 1 dye molecule per subunit and the image series used to construct the histogram was recorded with a shorter exposure time of 33 ms. The histogram is shown overlaid with fits to binomial distributions A) assuming ID complexes are well separated and B) allowing two complexes to be closer together than the resolution limit.

Table 5.5. Fitting parameters for the binomial distributions used to describe the histograms of single molecule photobleaching steps shown in Figure 5.13. n: number of subunits in a complex, θ : probability of detecting a fluorescent step for a particular fluorescent label, A: proportion of complexes which do not overlap.

Type of fit	n	θ	A	Mean square weighted deviation (χ^2_{red})
Well separated complexes	4	0.44 ± 0.02	-	3.2
Well separated complexes	5	0.36 ± 0.02	-	2.6
Overlapping complexes	2	0.65 ± 0.03	0.57 ± 0.06	2.2
Overlapping complexes	3	0.53 ± 0.03	0.83 ± 0.07	1.9

On examination of the values for θ shown in Table 5.4 and Table 5.5, it can be seen that the fluorophore detection rates emerging from the binomial fits lie roughly around 50 %. This is unexpectedly low considering the fact that an organic dye has been used as the fluorescent label, with average labelling efficiencies of 1 dye molecule per subunit. Further insight can be gained from modelling the distributions. Figure 5.14 A and B displays the results of modelling the two sets of Cy3-labelled ChID data shown in Figure 5.12 and Figure 5.13 respectively. The fluorophore lifetime (τ) was extracted from a dataset by summing all of the extracted intensity-time traces and fitting an exponential decay curve to the result. The unitary step size was determined from the average value produced during step counting, and the standard deviation of the noise in the intensity was calculated individually for each identified trace. Using these parameters extracted from the real data, a simulation was run to generate stepwise photobleaching traces in which a 2 state Markov model was employed for each fluorophore with the rate constant $1/\tau$ determining the transition from

the fluorescent state to the photobleached state. The simulated intensity-time traces were subjected to the same step-counting analysis as the real data to yield the displayed distributions. These data were modelled assuming that a complex consists of 3 subunits and that two complexes have a 15 % and a 17 % chance of overlapping for the data in panel A and B respectively. (Since all of the fits to the data appear similar, modelling the data using the underlying assumptions of any one of the fits should give a similar distribution to the real data).

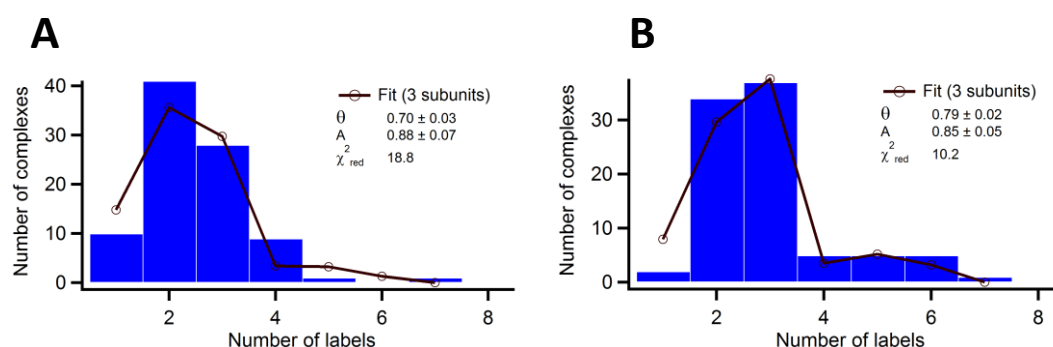


Figure 5.14. Simulated data of complexes comprising 3 subunits based on the fluorophore lifetimes and SNRs of the data shown in A) Figure 5.12 with a 15 % chance that two complexes will overlap, and B) Figure 5.13 with a 17 % chance that two complexes will overlap.

It can be seen immediately that the simulated distributions differ substantially from the original data, despite the fact that the assumptions used in the simulation were drawn from one of the fits to the original data. This suggests that the structure underlying the real data might not conform to any of the binomial models used to fit it. The values of θ derived from fits to the simulated data give an indication of the data quality and the performance of the step counting algorithm (experimental effects such as fluorescence quenching and pre-experiment photobleaching not having been included). Given the relatively poor signal to noise ratios of the original data (stemming from the difficulty of balancing fluorophore signal, fluorophore lifetime and substrate fluorescence), it is likely that the low values for θ reflect the probabilities of missing events due to swamping by noise in some cases, and due to events occurring close enough together in time that they appear as one step. These values are nevertheless greater than those derived from the same fit to the real data (0.7 and 0.79 compared to 0.56 and 0.53 respectively), meaning that there is a 20 % and 33 % chance respectively of missing a fluorophore due to experimental factors not accounted for such as fluorescence quenching and pre-experiment photobleaching if the simulated data and the original data are to be reconciled.

Since samples were protected from light before imaging, and data recording was begun before sample illumination (once the microscope had been focussed, a fresh field of view was used for photobleaching), it is unlikely that pre-experiment photobleaching could account for such high probabilities of missing a fluorophore. Cy3 has been used successfully previously in single molecule photobleaching studies for localization precision quantification[119] and stoichiometry determination[68] without interference from fluorescence quenching. The discrepancy between the real and simulated data might

therefore be due to the fact that the observed complexes comprise different numbers of subunits. All of the binomial fits discussed have assumed the underlying complex to consist of a fixed number of subunits. If the underlying complex population is heterogeneous, the only conclusion which can be drawn is that the number of ChID subunits forming a complex with ChII varies, and any particular complex can contain between 1 and 5 ChID subunits. It is not possible to directly interpret the experimental distributions as subunit numbers per complex due to limitations inherent in the experimental design which reduce the probability of observing each fluorescent label to less than 1. The relative proportions of complexes containing a certain number of subunits cannot be deduced from the data with any certainty. It is also not possible to say whether all of the potential subunit configurations represent an active complex.

This limitation is a feature of all single molecule stoichiometry work[84]. There is a trade-off to be made between the 1:1 labelling efficiency achieved by the use of fluorescent proteins with their higher instance of non-fluorescence, and more reliable organic dyes with the uncertainty of exact labelling efficiency. In order to satisfactorily address instances where a heterogeneous population of complexes exists, highly reliable labelling must be combined with reliable fluorophores. In the current work, experimental factors including varying labelling efficiency, substrate fluorescence and the unreliability of achieving TIRF with the experimental set-up in Figure 3.3 have limited the conclusions which can be drawn from the results.

Nevertheless, the conclusion that the number of ChID subunits associating into an ID complex is variable does not conflict with previous experimental evidence from titration experiments and analytical ultracentrifugation[17–19,25]. Optimum ratios of subunits obtained from titration experiments are not conclusive evidence that complexes with the stoichiometry such ratios suggest actually exist on the catalytic pathway, but it is striking that there is such variety in the ratios depending on which of the subunits are held in excess and limiting concentrations. The most consistent I:D stoichiometry emerging from titration experiments is 1:1 or 2:1. Combining this with the results from single molecule subunit counting suggests that ID complexes exist in a heterogeneous mix ranging from heterodimers (1 ChII and 1 ChID) to double pentamers. No evidence was found to support the presence of 6 ChID subunits in a complex, such as has been observed in the *R. capsulatus* cryo-EM structures[14,37]. This might be due to experimental conditions limiting the maximum number of fluorophores discernible in a complex. It has been noted by Ulbrich and Isacoff[79] that beyond 5 labels, the difficulty of detecting individual steps increases substantially and the label distributions for higher numbers of subunits begin to look very similar. However, the relatively low occurrence of higher label numbers in the current data (and lack of any 6-label complexes) suggests that if complexes containing 6 ChID subunits exist they occur in very small proportions compared to the majority of complexes. It is possible that the *R. capsulatus* magnesium chelatase subunits are anomalous in that a substantial proportion of ID complexes consistently form the observed double ring structure. It is the only ID complex which has been successfully isolated and subjected to detailed structural analysis via a technique which relies on averaging of many individual particles, thus potentially biasing the results in favour of the most prevalent complex form. In order to form the 3D reconstruction of the ATP-induced complex, 30 721

single particles were used. The largest homogeneous grouping within this population made up ~35 % of the total, and this fraction was used to produce the final reconstruction. Any other conformations present in the data set were not analysed.

Single molecule subunit counting experiments, while limited in the level of detail they can provide on exact proportions of complex populations, have demonstrated that the number of ChlD subunits present in the ID complex from *Synechocystis* varies. It is probable based on previous titration experiments that the number of ChlI subunits associating into the complex equals the number of ChlD subunits. Future work could extend subunit counting experiments to include labelled ChlI subunits. This would require the use of ChlI mutants since labelling wild type ChlI subunits was found to abolish chelatase activity, increasing the uncertainty of observing naturally occurring complexes.

6. Interaction of fluorescent nucleotide with the ID complex

ATP is required for the interaction between the I and D subunits, and the ATPase activity of the I subunit is necessary for the chelation reaction[11,19,29,32,120]. Previous work has investigated the ATPase activity and nucleotide binding kinetics of the ChII subunit in ensemble measurements[12,22]. While these studies have provided much insight into the behaviour of the ChII subunit and the role of its ATPase activity within the chelatase reaction, it was hoped to further expand this knowledge with direct observations of nucleotide binding to ChII and the ID complex at the single molecule level.

Single molecule studies of nucleotide binding dynamics necessitate the use of fluorescent nucleotide analogues, enabling tracking of nucleotide by fluorescence microscopy. Two fluorescent nucleotides have been used in the current study, 2'-(or-3')-O-(trinitrophenyl)adenosine 5'-triphosphate (TNP-ATP) and 2'/3'-O-(2-aminoethyl-carbamoyl)-adenosine-5'-triphosphate labelled with Cy3 (Cy3-ATP). The principle underlying single molecule fluorescent nucleotide binding experiments is distinguishing between bound nucleotide and nucleotide which is free in solution. Depending on the fluorescent nucleotide used, its fluorescence emission might depend heavily on its environment, or it could be relatively insensitive. In the former case, distinguishing a more strongly fluorescent bound nucleotide from free nucleotide should be relatively straightforward if the protein under investigation is immobilized. In the latter case, so long as the nucleotide concentration is low enough, the signal from bound, immobilized nucleotide will be distinguishable above the background fluorescence from nucleotide in solution.

6.1. TNP-ATP

TNP-ATP is a nucleotide derivative which has been modified through the ribose hydroxyls to carry a fluorescent label (Figure 6.1). It exists as a mixture of rapidly inter-converting 2' and 3' isomers due to the TNP group cross-linking the oxygens (Meisenheimer compound). The spectroscopic properties of TNP-ATP are highly environment-dependent[121,122]. While it has a very low fluorescence quantum yield in aqueous solution[122], its fluorescence emission has been observed to increase substantially upon binding to proteins[121–123]. This behaviour has led to the widespread use of fluorescent TNP-ATP analogues as reporters of ATP binding and structural probes within enzymes[121–125]. TNP-ATP has been used previously to probe the ATP binding kinetics of ChII[12]. Transient state kinetic measurements revealed that ChII undergoes Mg^{2+} -dependent isomerization (change in conformation) before binding nucleotide, the isomerization being slow relative to ligand binding. The increase in the fluorescence emission of TNP-ATP upon binding to ChII is shown in Figure 6.2. It can also be seen that the fluorescence emission maximum blue shifts slightly from 551 nm to 541 nm.

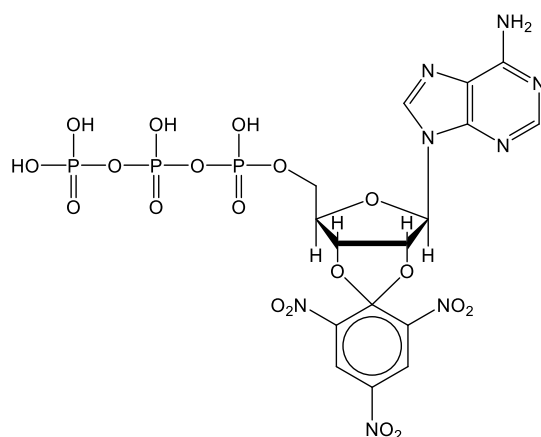


Figure 6.1. Chemical structure of TNP-ATP

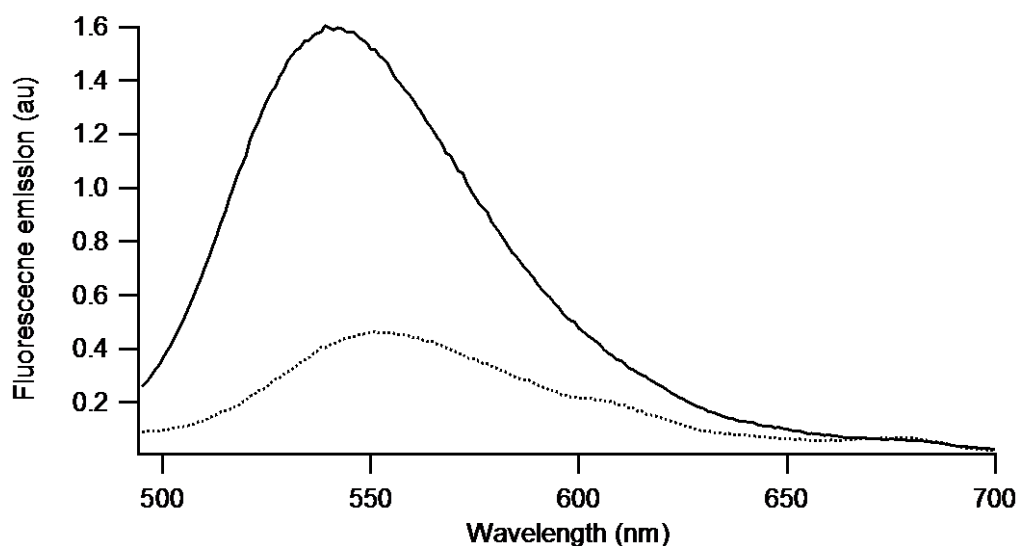


Figure 6.2. The fluorescence emission of TNP-ATP increases upon binding to ChII.

4.3 μM NHis₆-ChII was mixed with 5 μM TNP-ATP in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 °C) with 10 mM free Mg²⁺ and incubated for 5 min before recording the fluorescence emission spectrum with excitation at 408 nm, 5 nm slit width and 0.2 s integration time. Dotted line – TNP-ATP only, solid line – TNP-ATP with ChII.

TNP-ATP has been used previously in single molecule visualization of nucleotide binding[45]. TNP-ATP was observed while bound to single Klenow fragments of DNA polymerase I spread on a coverslip using total internal reflection (TIR) microscopy with laser excitation at 400 nm. Bound nucleotide was distinguished from nucleotide-free protein and unbound TNP-ATP which both lacked observable fluorescence emission.

TNP-ATP was chosen in the first instance for single molecule nucleotide binding studies because it had been used previously in single molecule studies[45]. Its environment-sensitive spectral properties are well suited to single molecule nucleotide binding observations, having low fluorescence emission when free in aqueous solution and increased fluorescence emission on binding to protein which greatly reduces the interference of background fluorescence from free nucleotide. Reid et al. had shown

previously that TNP-ATP bound to ChII through transient kinetic experiments[12], and single molecule binding experiments would build on this work to provide a more complete characterization of the nucleotide binding kinetics of this subunit.

6.1.1. TNP-ATP binds to ChII and can sustain an ID interaction

Earlier experiments have suggested that the fluorescent nucleotide analogue TNP-ATP competes with the natural substrate ATP for the binding site in ChII[12]. Millimolar quantities of ATP were required to displace TNP-ATP, and the dissociation constant (K_d) found for TNP-ATP binding to ChII in static binding experiments was $3.7\ \mu\text{M}$ at $2\ \text{mM}\ \text{Mg}^{2+}$. Figure 6.3 presents the results from a titration experiment to determine the dissociation constant for TNP-ATP binding to ChII at $10\ \text{mM}$ free Mg^{2+} . TNP-ATP was kept at a constant concentration of $2\ \mu\text{M}$ while the concentration of ChII was varied between $0\ \mu\text{M}$ and $30\ \mu\text{M}$. The change in the fluorescence emission of TNP-ATP excited at $408\ \text{nm}$ is shown in panel A. Panel B shows the change in the total fluorescence intensity (the integrated area under the emission spectrum) against protein concentration. Fitting Equation 2.2, describing a single site binding model, to the data in Figure 6.3 B yields a value of $(3.5 \pm 1.1)\ \mu\text{M}$ for K_d .

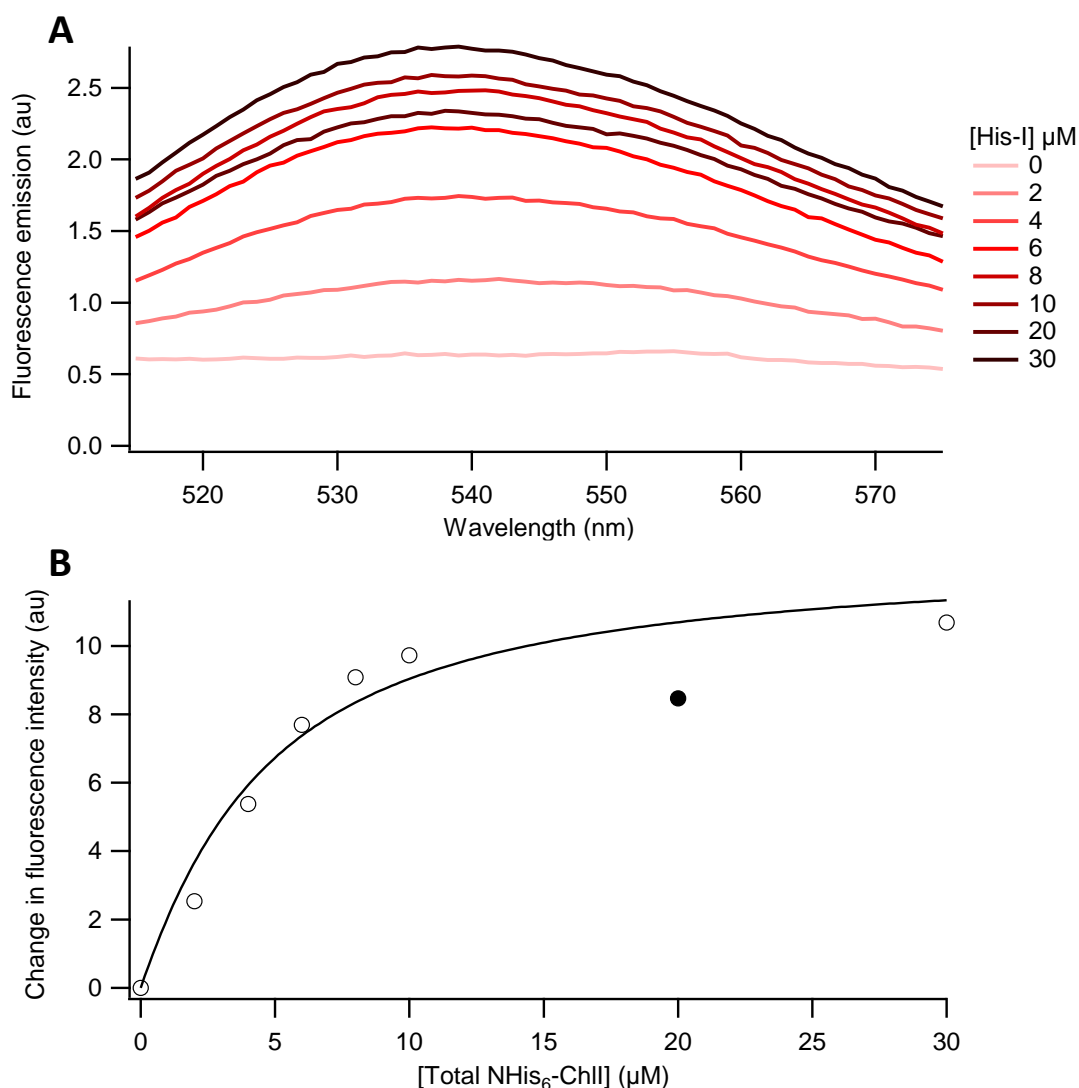


Figure 6.3. Titration assay to determine the dissociation constant (K_d) for TNP-ATP binding to NHis₆-ChII.

NHis₆-ChII was added to a solution of 2 μM TNP-ATP in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH 7.7 at 25 °C) with 10 mM free Mg^{2+} . The concentration of NHis₆-ChII was varied between 0 μM and 30 μM , and after a 5 min incubation at 25 °C the fluorescence emission from TNP-ATP was recorded with excitation at 408 nm for each independent titration point. A) Fluorescence emission spectra of TNP-ATP with varying concentration of NHis₆-ChII. The area under each fluorescence spectrum was calculated, and the change in fluorescence was plotted against NHis₆-ChII concentration (B). All data points are shown. The line shows a nonlinear fit to the data (excluding the point indicated by a filled circle) assuming a single site binding model with parameters (\pm standard deviation) $K_d = (3.5 \pm 1.1) \mu\text{M}$ and $\Delta F_{\text{max}} = (13 \pm 1) \text{ au}$.

The gel displayed in Figure 6.4 provides evidence that TNP-ATP is also capable of sustaining the ID complex. His₆-tagged ChID was mixed with non-tagged ChII and a slurry of Ni-NTA agarose beads. The His₆-tagged ChID subunits bound to the Ni-NTA and so remained in the mixture when the beads were subsequently washed several times in binding buffer (50 mM MOPS/NaOH, 0.1 M NaCl, 0.3 M glycerol, pH 7.7 at 25 °C). The final mixture was subjected to SDS-PAGE to determine which subunits were present. It can be seen from the gel that in the absence of nucleotide, ChII did not bind to ChID and so was removed from the mixture during the washing steps. When either ATP or TNP-ATP was present in the slurry and the

binding buffer used in the washing steps, however, ChII was observed on the gel, indicating that both of these nucleotides sustain the interaction between the I and D subunits.

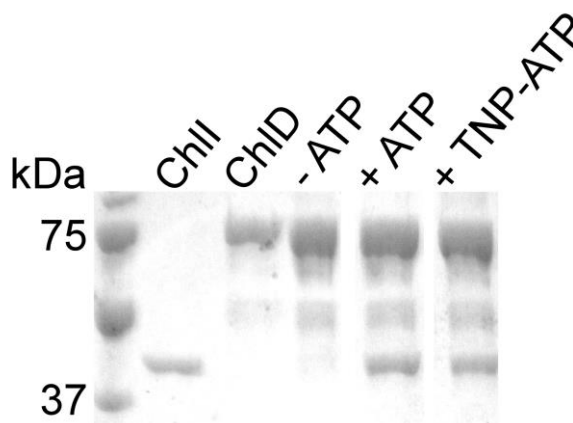


Figure 6.4. Co-purification of ChII and His₆-ChID in the presence of TNP-ATP.

5 μ M ChII was incubated for 10 min at room temperature with 5 μ M His₆-ChID, 10 mM free Mg²⁺ and either 5 mM ATP, 100 μ M TNP-ATP, or no nucleotide, in imidazole binding buffer (50 mM MOPS/NaOH, 0.1 M NaCl, 0.3 M glycerol, pH 7.7 at 25 °C) containing Ni-NTA agarose beads. The beads were then washed several times with imidazole binding buffer containing the appropriate nucleotide before the protein was eluted by the addition of SDS-PAGE gel-loading buffer and heating for 10 min at 90 °C.

6.1.2. TNP-ATP is an inhibitor of the chelatase reaction

Since it had been demonstrated that TNP-ATP both binds to ChII and facilitates formation of the ID complex, but had not been observed to be hydrolysed by ChII[12], the effect of TNP-ATP on the chelatase reaction was investigated. As with the use of fluorescently-labelled protein, using a fluorescent nucleotide in the chelatase assay runs the risk of interfering with the fluorescent detection of the product MgD_{IX}. Standard curves were constructed in the usual way, with the addition of different concentrations of TNP-ATP, shown in Figure 6.5. The absorption spectrum of TNP-ATP in particular overlaps with the excitation wavelength used for MgD_{IX}. This is evident from the reduced fluorescence intensity of MgD_{IX} as the concentration of TNP-ATP is increased. The fact that all of the standard curves converge to the same point at 0 μ M MgD_{IX} implies that while the absorption of the TNP-ATP is reducing the excitation light available to MgD_{IX}, any fluorescence emission from TNP-ATP bound to enzyme is below the detection limit of the platereader. Therefore all observed fluorescence emission is due to MgD_{IX}.

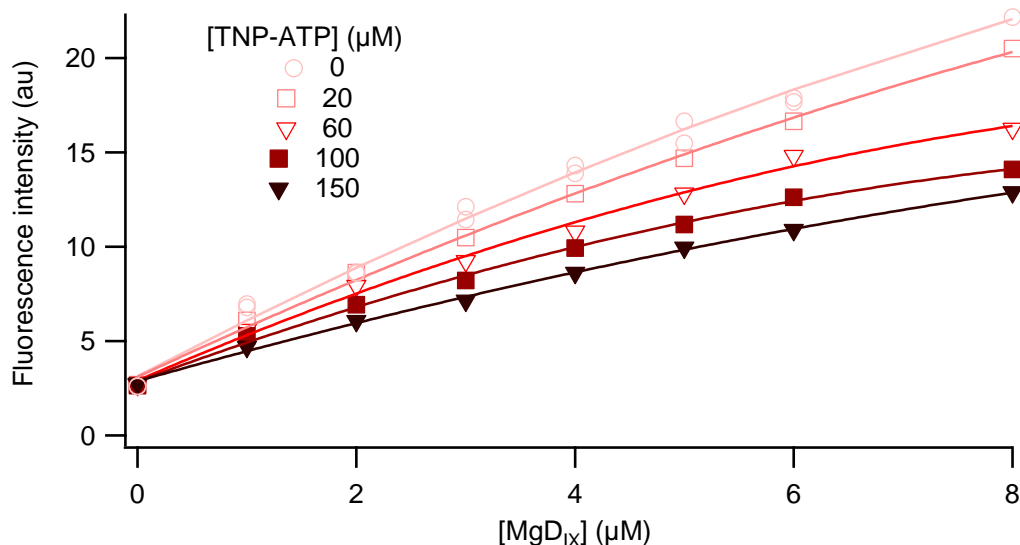


Figure 6.5. MgD_{IX} standard curves in the presence of TNP-ATP.

Standards were prepared in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 34 °C) containing chelatase subunits (0.1 μM ChII, 0.1 μM ChID and 0.4 μM ChIH) and TNP-ATP at the concentrations indicated in the graph legend. All data points are shown. The lines indicate 2nd order polynomial fits to the data.

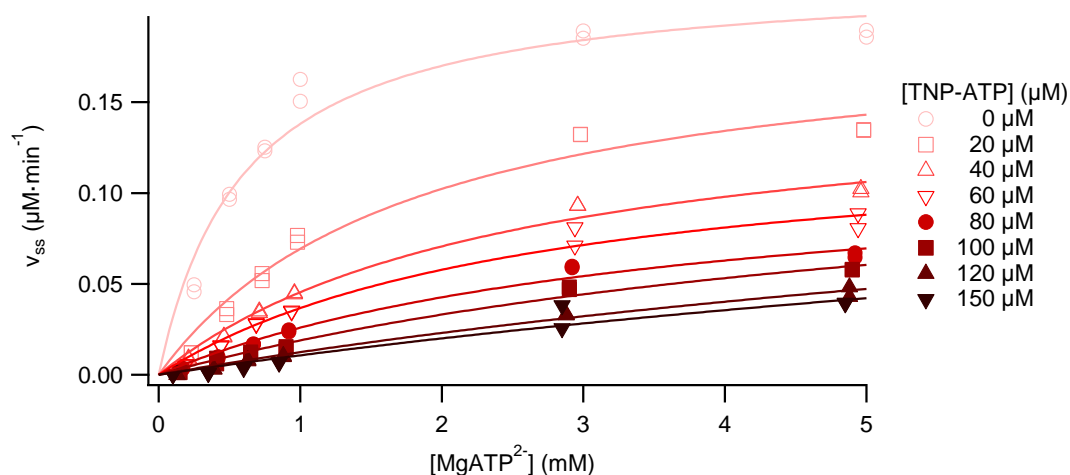


Figure 6.6. Plots of steady state rates against substrate concentration for chelatase assays in the presence of TNP-ATP.

Chelatase assays were run in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH7.7 at 34 °C) with 10 mM free Mg^{2+} and standard subunit concentrations (0.1 μM ChII, 0.1 μM ChID, 0.4 μM ChIH). The concentration of ATP was varied between 0.25 mM and 5 mM, and TNP-ATP concentration was varied between 0 and 150 μM . The lines indicate fits to the Michaelis Menten equation (Equation 2.1).

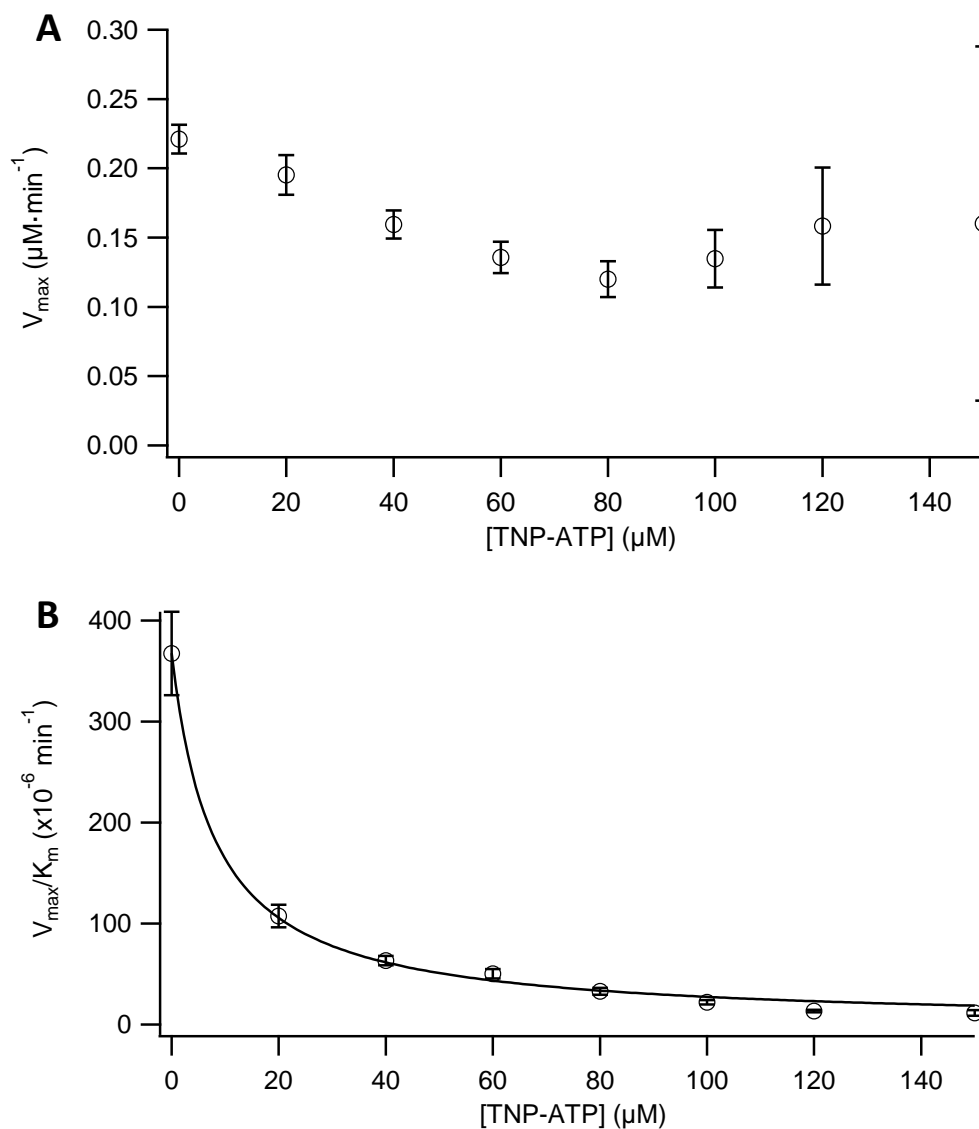


Figure 6.7. Plots of A) V_{\max} and B) V_{\max}/K_m against concentration of TNP-ATP for chelatase assays.

The apparent values of V_{\max} and V_{\max}/K_m were obtained from the fitted curves shown in Figure 6.6. Error bars show standard errors. The line indicates a fit to Equation 6.1.

Magnesium chelatase assays were performed at 6 different concentrations of ATP in the range 0.25 mM to 5 mM. These were repeated in the presence of TNP-ATP varying in concentration from 0 μM to 150 μM. Individual calibration curves were used at each concentration of TNP-ATP to convert the recorded fluorescence intensity to the corresponding MgD_{IX} concentration. The resulting plots of steady state rate against ATP concentration for different concentrations of TNP-ATP are shown in Figure 6.6. These were fitted with the Michaelis Menten equation (Equation 2.1) to yield values for V_{\max} and V_{\max}/K_m which are displayed in Figure 6.7 for each concentration of TNP-ATP, indicating mixed inhibition. Values of V_{\max}/K_m are reliable across the range of TNP-ATP concentrations. In contrast V_{\max} is only well defined by these data at concentrations of TNP-ATP less than approximately 60 μM so it was not fitted. Fitting the data for V_{\max}/K_m to Equation 6.1 yielded a value for the dissociation constant of TNP-ATP for the chelatase complex of $(8.1 \pm$

0.5) μM . The fact that TNP-ATP exhibits mixed inhibition of the chelatase reaction suggests that as well as competing for the ATP binding site in ChII, it binds elsewhere on the chelatase complex.

$$\frac{V^{obs}}{K} = \frac{(V/K)_{lim}}{1 + \frac{[I]}{K_i}}$$

Equation 6.1. Relationship between V/K and inhibitor concentration

6.1.3. Single molecule visualization of TNP-ATP binding to chelatase subunits

The binding of TNP-ATP to ChII and to the ID complex was investigated at the single molecule level via fluorescence microscopy. ChII, or ChII and His₆-tagged ChID subunits were mixed in the presence of TNP-ATP and immobilized on a Ni-NTA functionalized coverslip. Any TNP-ATP bound to the immobilized protein should then be visible under 405 nm laser illumination, with unbound fluorescent nucleotide producing negligible background fluorescence. ChID subunits were fluorescently labelled with Cy5 in some cases to enable localization of the complexes.

Protein was immobilized on coverslips to give approximately 150 complexes in a 30 μm by 30 μm field of view. The total surface area available for immobilization was approximately 8 mm by 15 mm, and the volume of the sample chamber was roughly 10 μl , meaning the concentration of protein in the sample chamber was about 3.3 pM. Using the dissociation constant for TNP-ATP for the ID complex of 8.1 μM , the fraction of protein which will have TNP-ATP bound at any particular time can be calculated according to

$$\frac{EL}{E_t} = \frac{E_t + L_t + K_d - \sqrt{(E_t + L_t + K_d)^2 - 4E_tL_t}}{2E_t}$$

Equation 6.2. Fraction of ligand-bound protein in terms of total enzyme and ligand concentrations

where EL is the concentration of ligand-bound enzyme, E_t is the total enzyme concentration and L_t is the total ligand concentration. With TNP-ATP at a concentration of 10 μM , the fraction of protein with TNP-ATP bound is 0.552, or 83 out of 150 complexes in a field of view.

Figure 6.8 shows examples of surface-immobilized ID complexes in the presence of TNP-ATP. Although the Cy5-labelled ChID is clearly visible under 647 nm illumination (panel A), bound TNP-ATP cannot be discerned under 405 nm illumination (panel B). Approximately 60 % of the complexes visible under 647 nm illumination would be expected to have TNP-ATP bound. Panels C and D show another example where unlabelled complexes were immobilized at 4 times the concentration shown in panel A and mixed with 20 μM TNP-ATP. This was to ensure that the label on the ChID subunits was not interfering with nucleotide binding (though it has been shown that labelled subunits can still interact with each other). Under these conditions it can be expected that 74 % of complexes will bind

TNP-ATP, or approximately 446 fluorescent spots should be visible under 405 nm illumination. However, only a few fluorescent clumps are visible in panel C, and it is likely that these are due to aggregation of TNP-ATP around impurities on the surface, or intrinsic surface fluorescence, since similar clumps are visible in the blank surface (TNP-ATP only) shown in panel D.

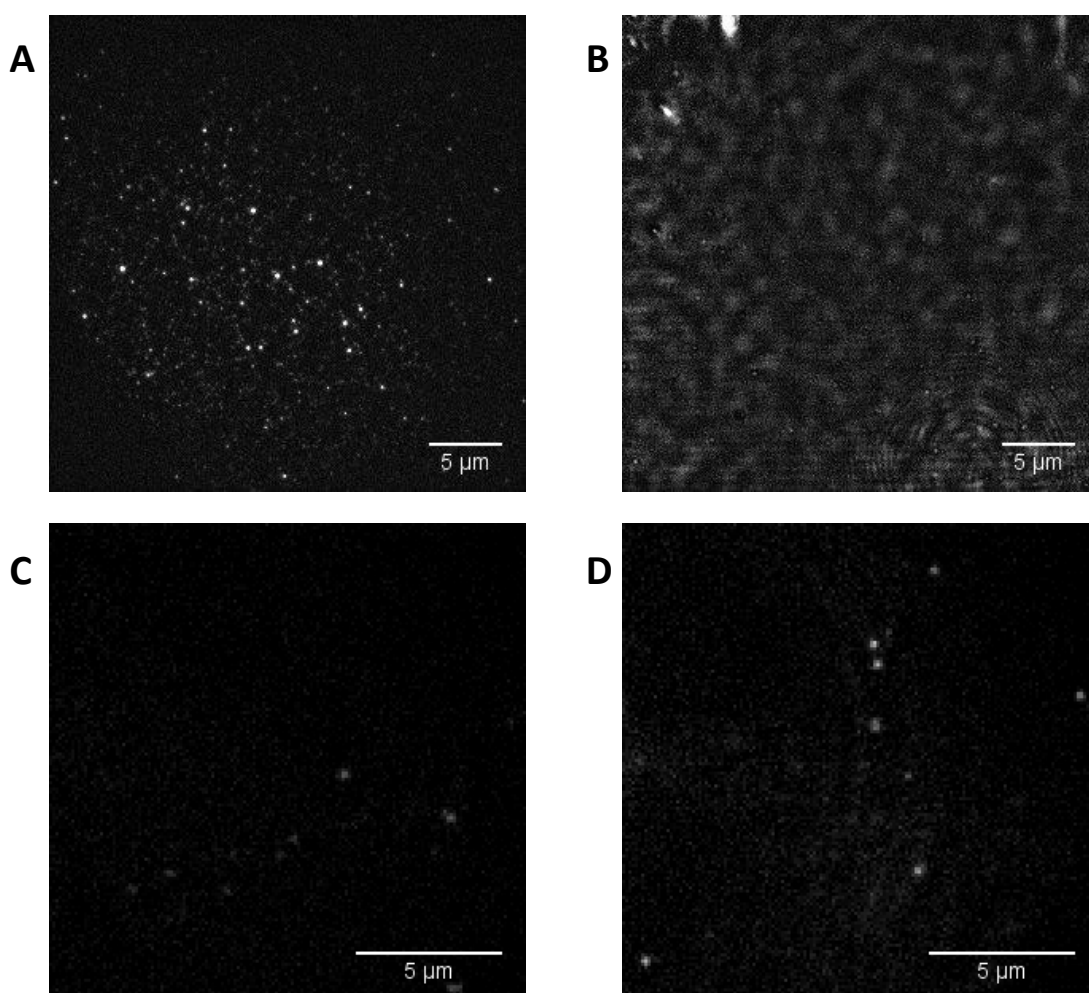


Figure 6.8. ID complexes immobilized on a coverslip in the presence of TNP-ATP.

1 μM ChII was incubated with 1 μM His₆-ChID-Cy5 for 5 min in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, 1 mM DTT, pH 7.7 at 25 °C) with 10 mM free Mg²⁺ and 10 μM TNP-ATP before rapid dilution to a concentration of 25 pM and immobilization on a coverslip. Images A) and B) show the same sample under 0.58 mW 647 nm laser illumination and 0.07 mW 405 nm laser illumination respectively. Image C) shows unlabelled ID complexes immobilized at a concentration of 100 nM in chelatase buffer with 20 μM TNP-ATP and D) a blank coverslip with 20 μM TNP-ATP, both illuminated with a 405 nm laser. Images A) and B) were produced from the sum of 10 original images recorded with a 0.1 s exposure in TIR mode and subjected to rolling ball background removal with a radius of 10 pixels. Images C) and D) were produced from the sum of 5 original images recorded with a 1 s exposure and subjected to rolling ball background removal with a radius of 10 pixels.

It was found that under illumination at 405 nm, there is a lot of interference from reflections in the oil and extraneous fluorescence which made the detection of bound TNP-ATP impossible. This is in contrast to a previous study in which TNP-ATP was observed binding to single Klenow fragments of DNA polymerase I[45]. In this study, however,

nucleotide-protein complexes were spread and dried onto a coverslip before observation at exposure times of 30 s. It was hoped that with the improvements in camera technology witnessed in the intervening years it would be possible to observe TNP-ATP binding in solution and in real time, enabling extraction of binding kinetics. TNP-ATP was chosen for the fluorescent nucleotide because it was already known to bind to ChII, and its spectroscopic properties recommended it for use as a single molecule probe (its low fluorescence emission in the unbound state reducing interfering background fluorescence). Unfortunately, the signal from bound TNP-ATP proved too weak to compete with the high levels of intrinsic fluorescence in the sample buffer and coverslip under 405 nm illumination, at least on the time scales required for real time imaging of binding events.

6.2. Cy3-EDA-ATP

Like TNP-ATP, Cy3-ATP is a fluorescent nucleotide analogue which has been modified through the ribose hydroxyls (Figure 6.9). Addition of the linker ethylenediamine enables introduction of an amine reactive fluorophore[126]. Cy3-ATP also exists as a mixture of 2' and 3' isomers, but these inter-convert much more slowly (re-equilibration takes a few hours at room temperature and neutral pH) so it is possible to separate and investigate the isomers individually. Cy3-ATP has a larger extinction coefficient than bound TNP-ATP and exhibits stronger fluorescence[94]. Its fluorescence emission is not as environmentally sensitive as that of TNP-ATP, meaning that binding and dissociation to and from protein does not cause a change in its fluorescence, nor does hydrolysis[92].

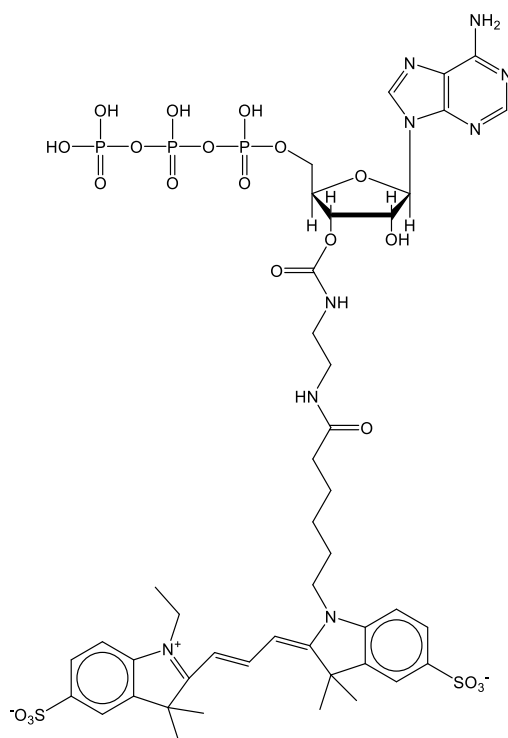


Figure 6.9. Chemical structure of Cy3-EDA-ATP

Cy3-ATP was selected for several reasons. The microscope set-up already contained the filters necessary for viewing Cy3 fluorescence. Although Cy3-ATP does not exhibit an increase in fluorescence upon binding like TNP-ATP or DEAC-ATP[127], its excitation

maximum (550 nm) is at a longer wavelength than that of DEAC-ATP (430 nm). As outlined in Section 6.1.3. the samples exhibited a high background fluorescence under shorter wavelength illumination which interfered with the fluorescence emission signal of interest. Cy3-ATP is also commercially available, so does not require synthesizing.

6.2.1. Separation of Cy3-EDA-ATP isomers

Cy3-ATP was separated into its two isomers via HPLC in TEAB buffer, following the method of Oiwa et al.[94] The nucleotide was detected spectroscopically through its absorbance at 260 nm. An example of an HPLC absorbance record displaying the separation is shown in Figure 6.10.

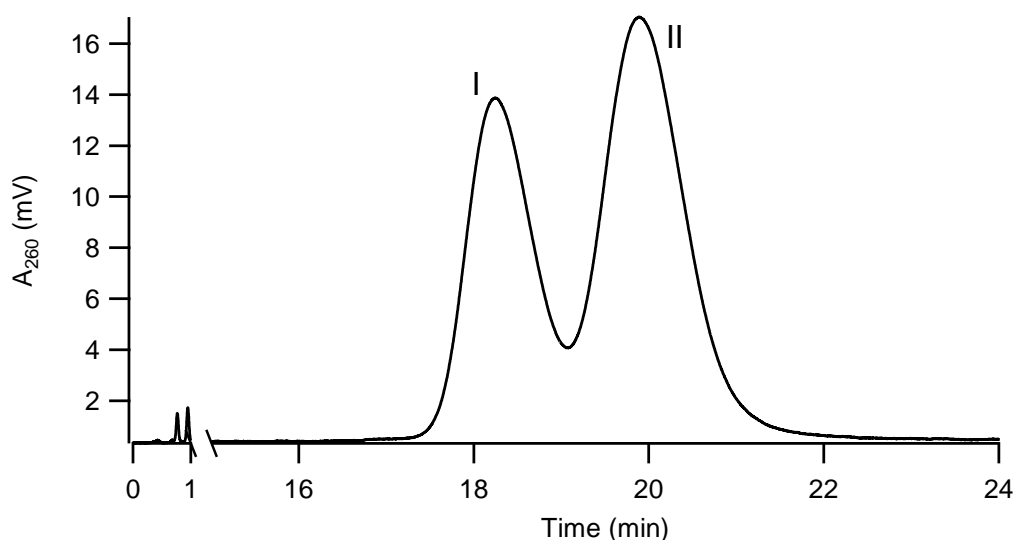


Figure 6.10. Separation of Cy3-EDA-ATP isomers via HPLC.

Cy3-ATP samples were separated into their two constituent isomers I and II (2'-O-Cy3-EDA-ATP and 3'-O-Cy3-EDA-ATP respectively) using a Nova-Pak C18 column by isocratic elution in TEAB buffer (100 mM TEAB, 12 % acetonitrile, pH 7.4 at 25 °C). An example of an HPLC absorbance record at 260 nm is shown.

Once purified, the two isomers were dried before being re-dissolved in water and stored at -80 °C until use. Several methods were attempted to dry the samples with varying success since exposing the isomers to elevated temperature or high pH causes the rate at which they re-equilibrate to increase. Samples were tested after drying using the same HPLC method to check the purity of the isomers. The result after evaporation of solvent *in vacuo* aided by the addition of methanol (method recommended by Oiwa) is displayed in Figure 6.11. This attempt to dry isomer I (2'-O-Cy3-EDA-ATP) took approximately 2 h. It is evident from the HPLC trace that essentially the entire sample re-equilibrated during this time.

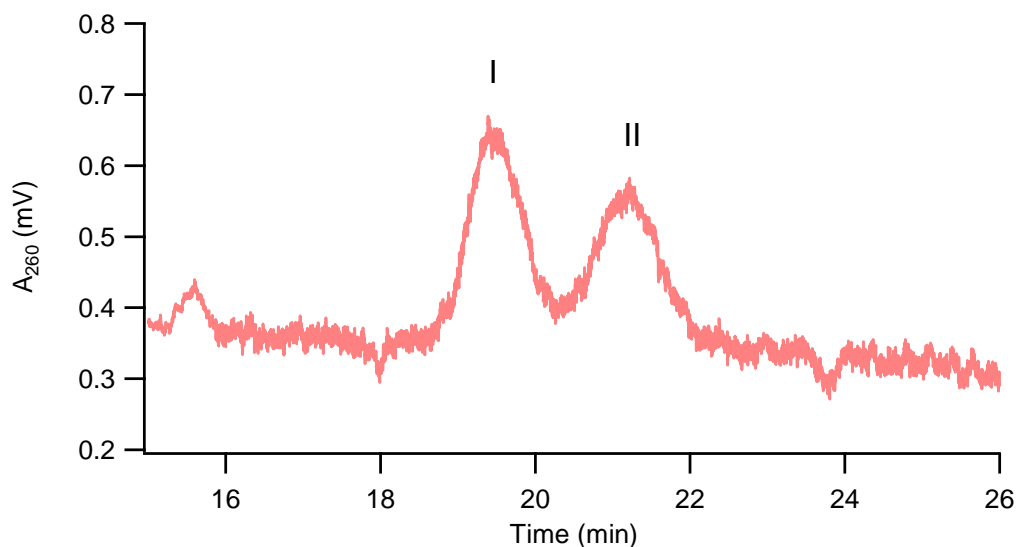


Figure 6.11. Purity of Cy3-EDA-ATP isomer I after evaporation with addition of methanol *in vacuo*.

Purified isomer I was dried on a rotary evaporator with addition of methanol to remove residual triethylamine. The final material was re-dissolved in water. The purity of the resulting solution was checked by HPLC under the same conditions used for purification (isocratic elution from a C18 column in TEAB buffer with detection by absorbance at 260 nm). The HPLC absorbance record is shown.

Figure 6.12 shows successful attempts at drying the samples while maintaining the purity of the isomers. Panel A displays the result of directly freeze drying isomer I while isomer II (3'-O-Cy3-EDA-ATP; panel B) was partially dried *in vacuo* with the addition of acetone and then freeze dried. The HPLC traces were fitted using the Multi-peak Fitting package available in Igor Pro 6.2 (WaveMetrics, Inc., Lake Oswego, OR, US), revealing the purity of isomers I and II to be 87 % and 85 % respectively.

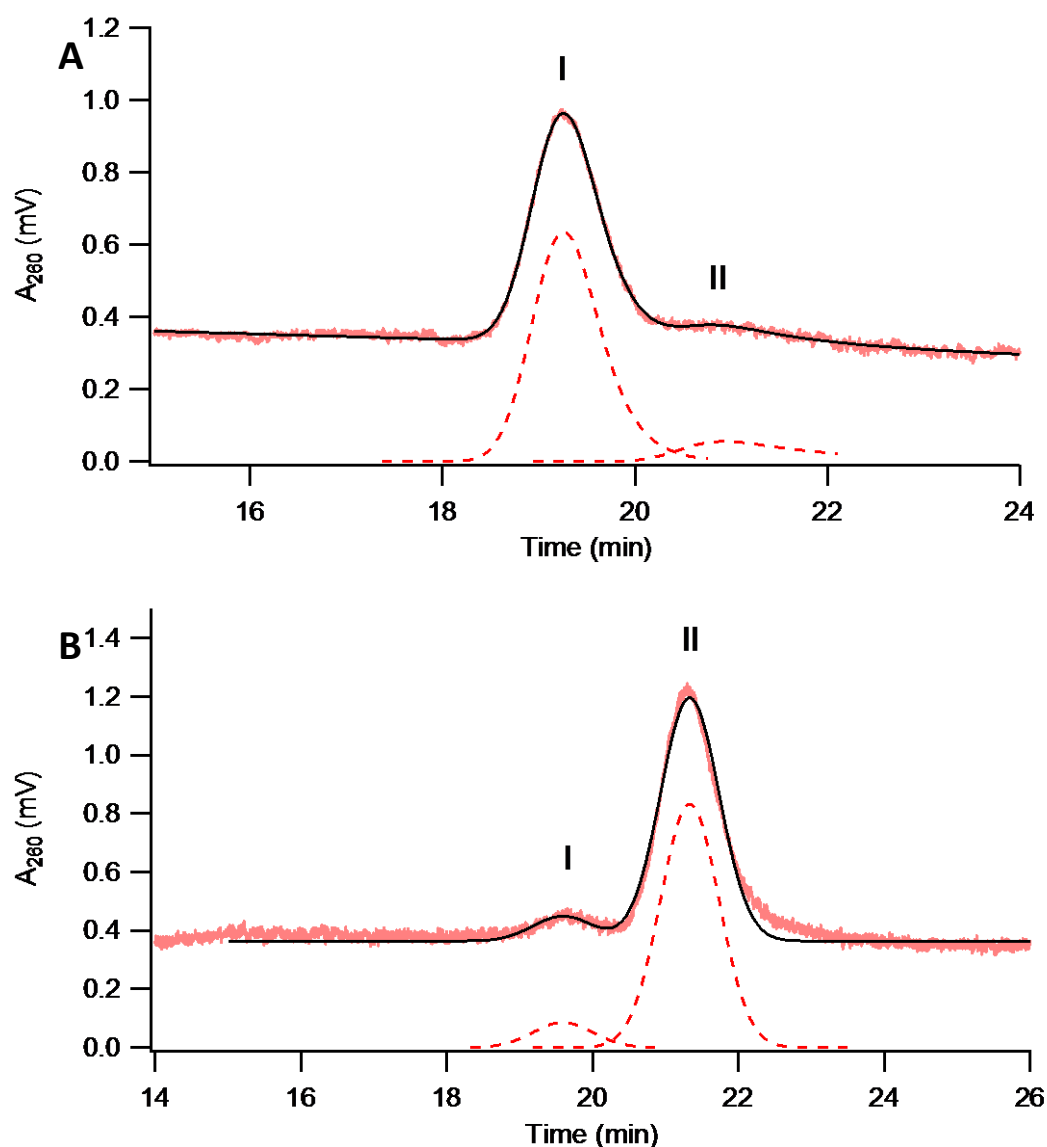


Figure 6.12. Purity of Cy3-EDA-ATP isomers after drying.

Purified isomer I was freeze-dried overnight then re-dissolved in water. Purified isomer II was partially dried on a rotary evaporator with addition of acetone and then freeze-dried before being re-dissolved in water. The purity of the resulting solutions was checked by HPLC under the same conditions used for purification (isocratic elution from a C18 column in TEAB buffer with detection by absorbance at 260 nm). The HPLC absorbance records are shown for isomer I (A) and isomer II (B). The peak fits shown indicate purities of 87 % and 85 % for isomer I and II respectively. Light pink line – original HPLC trace; black line – peak fit to HPLC trace; dashed line – individual peaks from peak fit offset from main trace.

The purified isomers were also tested to check whether the drying process had caused hydrolysis of the ATP. The samples were subjected to further analysis via HPLC using a phosphate buffer to separate Cy3-ATP from Cy3-ADP[94]. An example HPLC trace showing peaks corresponding to both isomers of Cy3-ATP and Cy3-ADP is reproduced in Figure 6.13. Cy3-ADP elutes after Cy3-ATP in this buffer system.

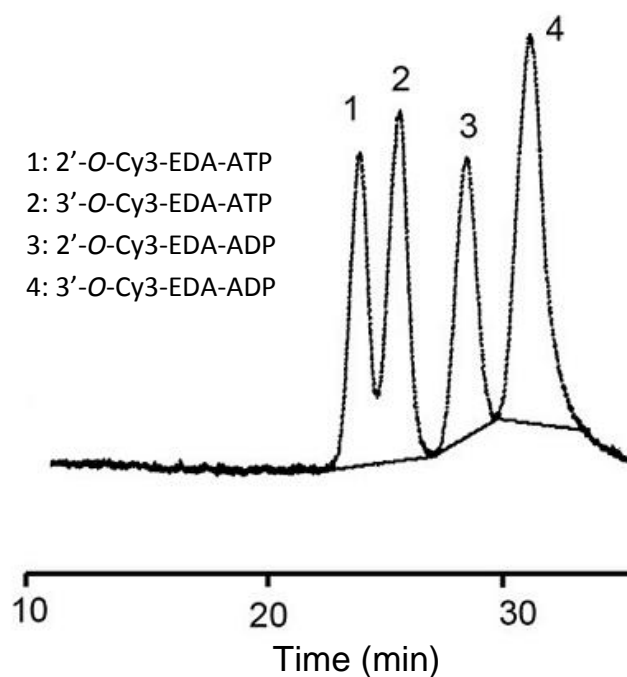


Figure 6.13. HPLC absorbance trace showing isomers of Cy3-EDA-ATP and Cy3-EDA-ADP.

Detection was by absorbance at 260 nm. This figure has been reproduced from an article in PNAS: Amrute-Nayak, M. et al. ATP turnover by individual myosin molecules hints at two conformers of the myosin active site (Supporting Information). *PNAS*. 111, 2536-2541 (2014). © National Academy of Sciences, USA.

Figure 6.14 presents the HPLC traces for purified isomer II. Some of the original, unpurified nucleotide was run to provide a marker indicating at what time Cy3-ATP would elute (panel A). Panel B displays the trace for a 1:1 mixture of original material and purified isomer II. The peak corresponding to isomer I remains constant while the peak for isomer II roughly doubles in size with the addition of the purified isomer. The fact that the HPLC traces for the original Cy3-ATP material and the purified isomer overlap completely and no peak is visible corresponding to Cy3-ADP indicates that the purified isomer is 3'-O-Cy3-EDA-ATP.

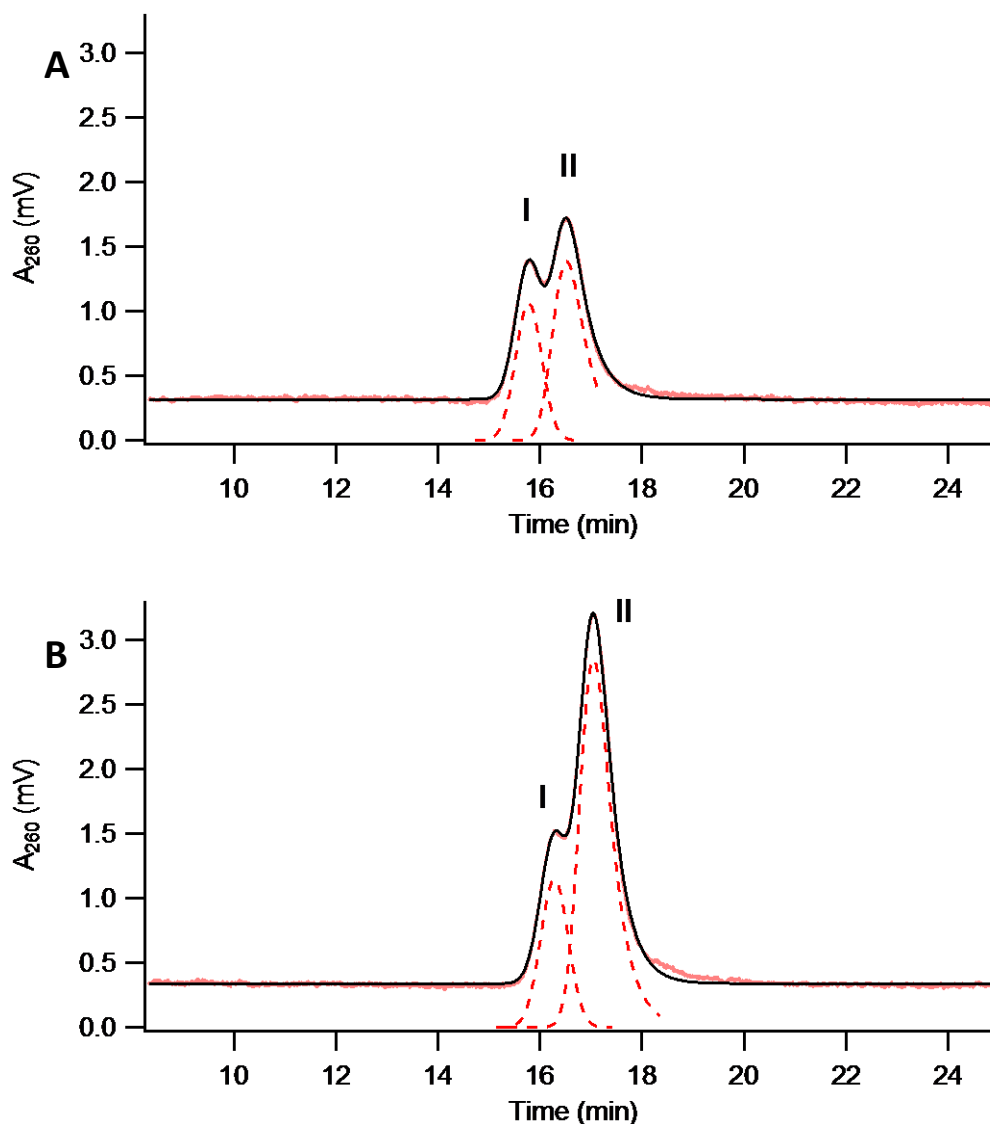


Figure 6.14. Check for presence of 3'-O-Cy3-EDA-ADP in purified 3'-O-Cy3-EDA-ATP.

Cy3-EDA-ATP was isocratically eluted from a Nova-Pak C18 column in phosphate buffer (10 mM $\text{KH}_2\text{PO}_4/\text{K}_2\text{HPO}_4$, 11 % acetonitrile, pH 6.8 at 25 °C) and detected by absorption at 260 nm. HPLC absorbance records are shown for A) 10 μM unseparated Cy3-EDA-ATP and B) a 1:1 mixture of 10 μM unseparated Cy3-EDA-ATP and 10 μM purified isomer II. Light pink line – original HPLC trace; black line – peak fit to HPLC trace; dashed line – individual peaks from peak fit offset from main trace.

6.2.2. Cy3-EDA-ATP binds to ChII

The question of whether Cy3-ATP binds to the nucleotide binding site of ChII was addressed through a competition experiment with TNP-ATP, the dissociation constant for TNP-ATP having already been determined as $(3.5 \pm 1.1) \mu\text{M}$. TNP-ATP was allowed to bind to ChII, and was then displaced by successive additions of Cy3-ATP. The concentration of bound TNP-ATP was monitored by its fluorescence emission. Figure 6.15 A shows the fluorescence emission spectra of the mixture as Cy3-ATP is added. It can be seen that although the sample is illuminated at 408 nm, Cy3-ATP contributes some fluorescence (the emerging peak at $\sim 560 \text{ nm}$). This is due to direct excitation of the Cy3 fluorophore, and not FRET between TNP and Cy3, since the emission spectra could be corrected for the Cy3

fluorescence (Figure 6.15 B: background spectra were recorded for Cy3-ATP only excited at 408 nm under the same conditions at the corresponding concentrations, and subtracted from the sample spectra).

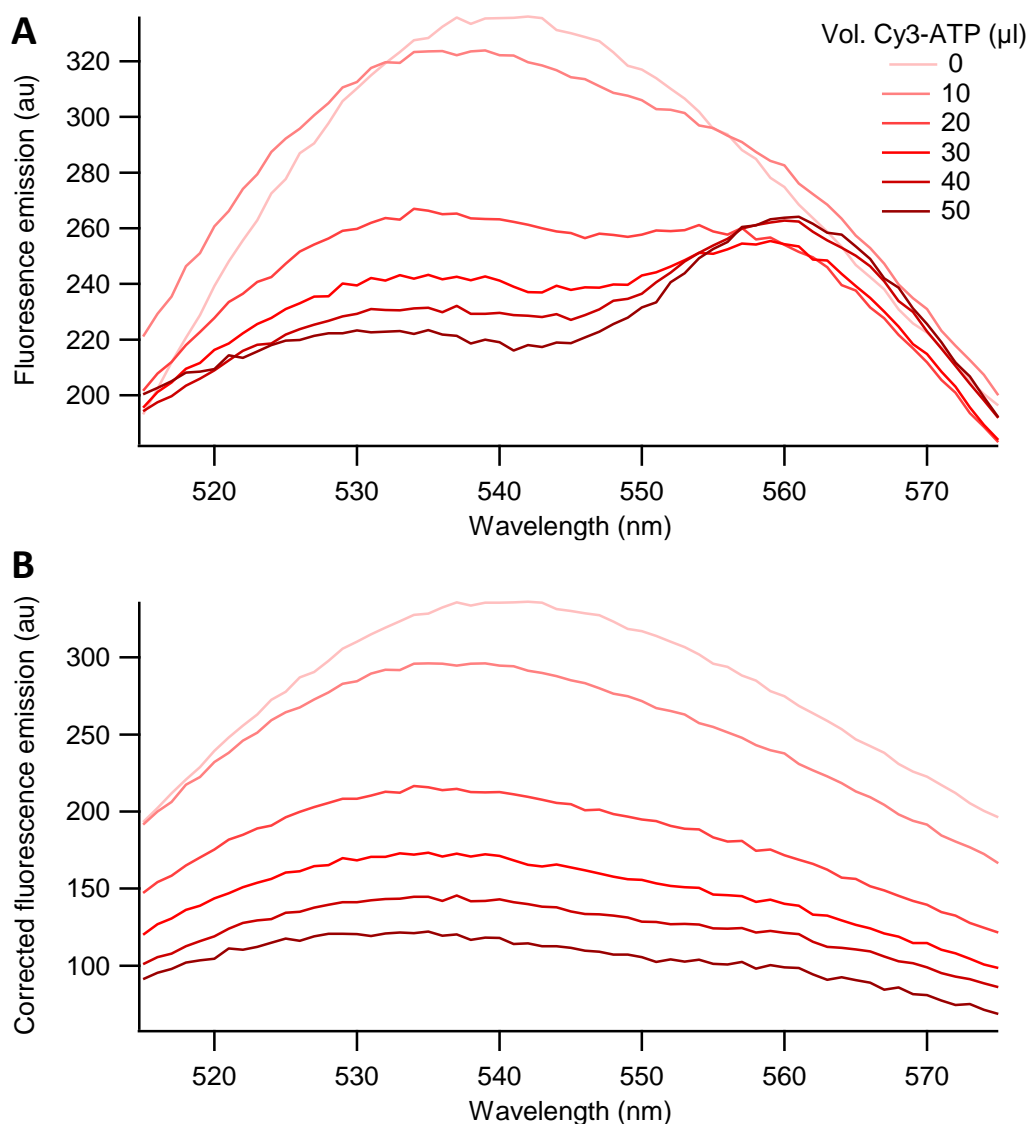


Figure 6.15. Fluorescence emission spectra for competition titration between TNP-ATP and Cy3-ATP in the presence of ChII.

10 μ l volumes of 10 μ M Cy3-ATP were added in succession to a starting volume of 100 μ l containing 5 μ M TNP-ATP and 5 μ M NHis₆-ChII in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 °C) with 10 mM free Mg²⁺. After each addition of Cy3-ATP the sample was incubated for 5 min at 25 °C before the fluorescence emission of TNP-ATP was recorded with excitation at 408 nm. A) Raw fluorescence emission spectra. B) Fluorescence emission spectra corrected for contribution from Cy3-ATP.

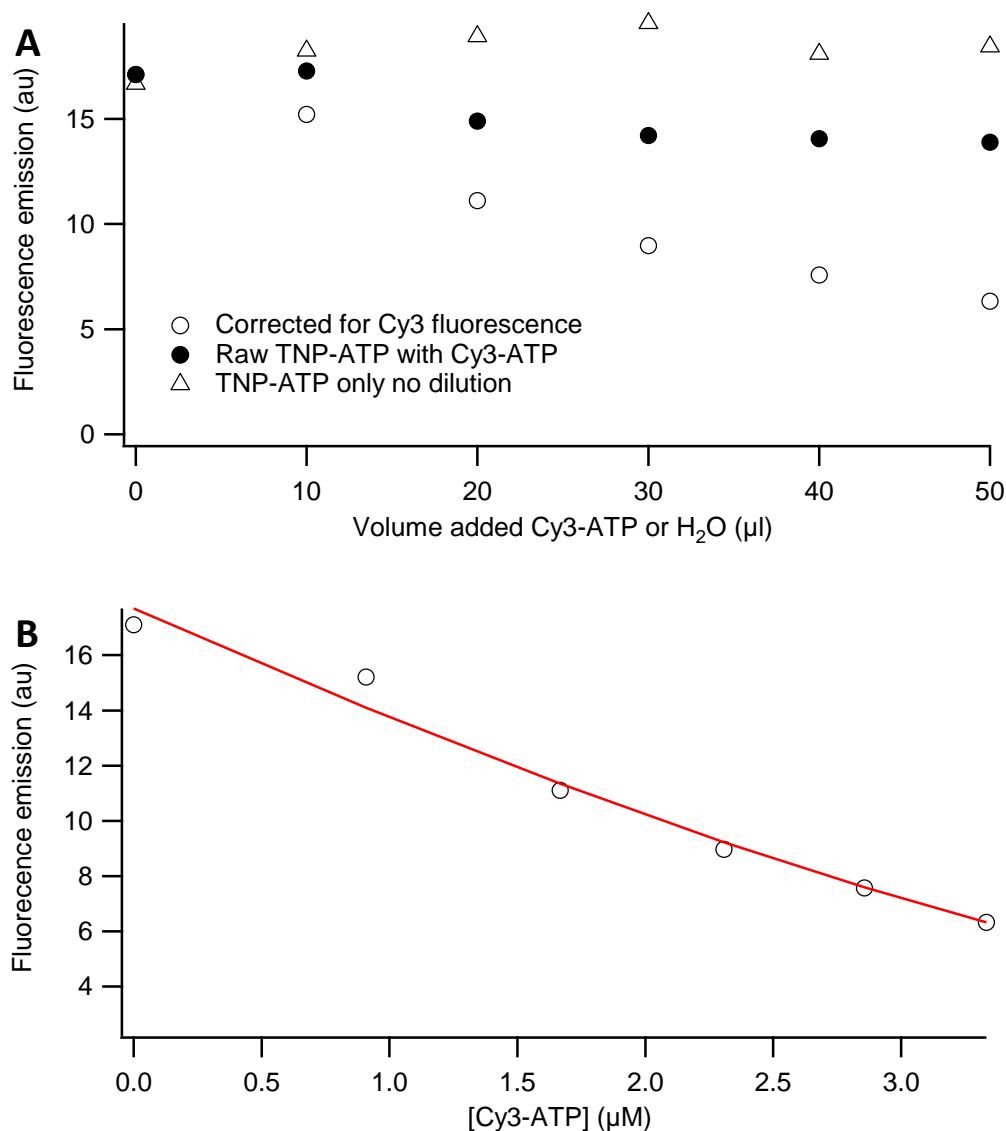


Figure 6.16. Total fluorescence emission for competition titration between TNP-ATP and Cy3-ATP in the presence of ChII.

10 μl volumes of 10 μM Cy3-ATP were added in succession to a starting volume of 100 μl containing 5 μM TNP-ATP and 5 μM NHis₆-ChII in chelataase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 °C) with 10 mM free Mg²⁺. After each addition of Cy3-ATP the sample was incubated for 5 min at 25 °C before the fluorescence emission of TNP-ATP was recorded with excitation at 408 nm. A) Variation of fluorescence emission with volume of Cy3-ATP added. B) Corrected fluorescence emission with a fit to the data yielding a value for $K_{d,Cy3-ATP}$ of (1.4 ± 0.4) μM.

The experiment was conducted in a single cuvette with successive additions of Cy3-ATP at constant volume due to the limited quantity of the nucleotide, so controls were also run to check that successive illumination of the sample was not significantly photobleaching the TNP-ATP and to compensate for the effect of dilution on the fluorescence emission. Figure 6.16 A shows the total fluorescence emission from successive titrations for the raw sample, corrected sample and the photobleaching control. The photobleaching control was made up with the starting concentration of TNP-ATP and protein and not subjected to dilution (TNP-ATP only no dilution). The fluorescence emission from this control remained constant throughout successive illumination cycles. DynaFit[107] is a program capable of fitting

competitive binding data in circumstances where the enzyme concentration is comparable to that of the competing ligands, taking into account dilution effects from subsequent ligand additions. It was therefore used to fit the dilution control and corrected sample data to the competitive binding mechanism shown in Scheme 2.1 yielding a dissociation constant for Cy3-ATP of $(1.4 \pm 0.4) \mu\text{M}$ (Figure 6.16 B). The uncertainty associated with this value is relatively large, however it can be seen from simulations for $K_{d,\text{Cy3-ATP}}$ values over several orders of magnitude (Figure 6.17) that it is correct to an order of magnitude. This is accurate enough to determine that it does displace TNP-ATP and bind to ChII. An Inhibition assay was not conducted with Cy3-ATP due to the limited amount of Cy3-ATP available.

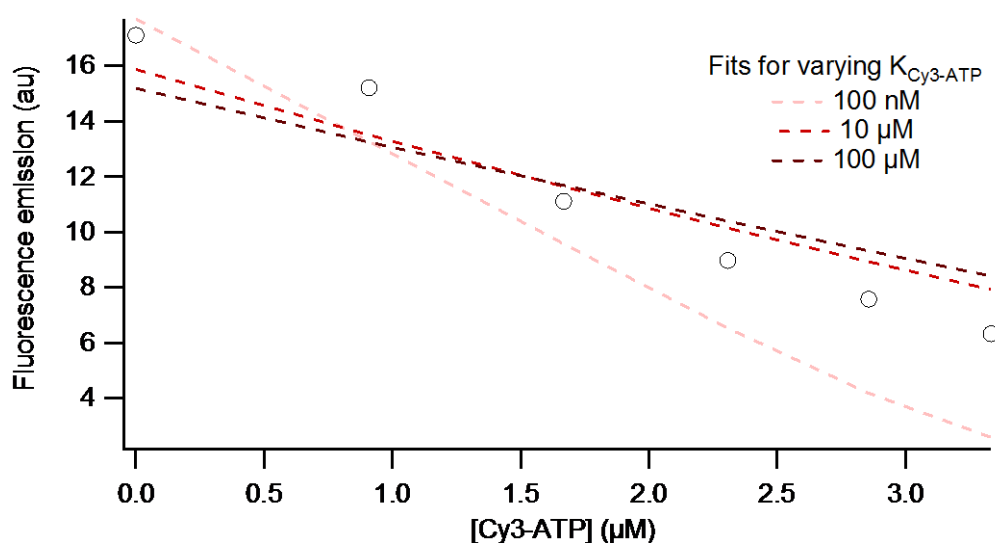


Figure 6.17. Simulations for varying $K_{d,\text{Cy3-ATP}}$. All models were generated using Dynafit[107] with the value for $K_{d,\text{TNP-ATP}}$ fixed at $3.5 \mu\text{M}$.

6.2.3. Single molecule visualization of Cy3-EDA-ATP interaction with His₆-ChII

The binding of Cy3-ATP to ChII was investigated at the single molecule level via fluorescence microscopy. His₆-tagged ChII was immobilized on a Ni-NTA functionalized coverslip and imaged in the presence of Cy3-ATP. All experiments were performed with 3'-O-Cy3-EDA-ATP, referred to as Cy3-ATP for convenience.

Since the fluorescence emission from Cy3-ATP remains constant whether unbound or bound to protein, free Cy3-ATP in solution contributes significantly to the background fluorescence in the sample. It is therefore necessary to employ TIR illumination to reduce the volume of the sample being illuminated as much as possible. This also restricts the maximum concentration of nucleotide which can be used in experiments so that the background fluorescence does not swamp the immobilized fluorophore signal. Figure 6.18 displays example images of Cy3-labelled ChID subunits immobilized on a coverslip in different concentrations of Cy3-ATP. These give an idea of how difficult it is to discern immobilized fluorophores against a fluorescent background. 60 nM Cy3-ATP was chosen as the highest concentration of free fluorescent nucleotide which still allowed most immobilized fluorophores to be identified.

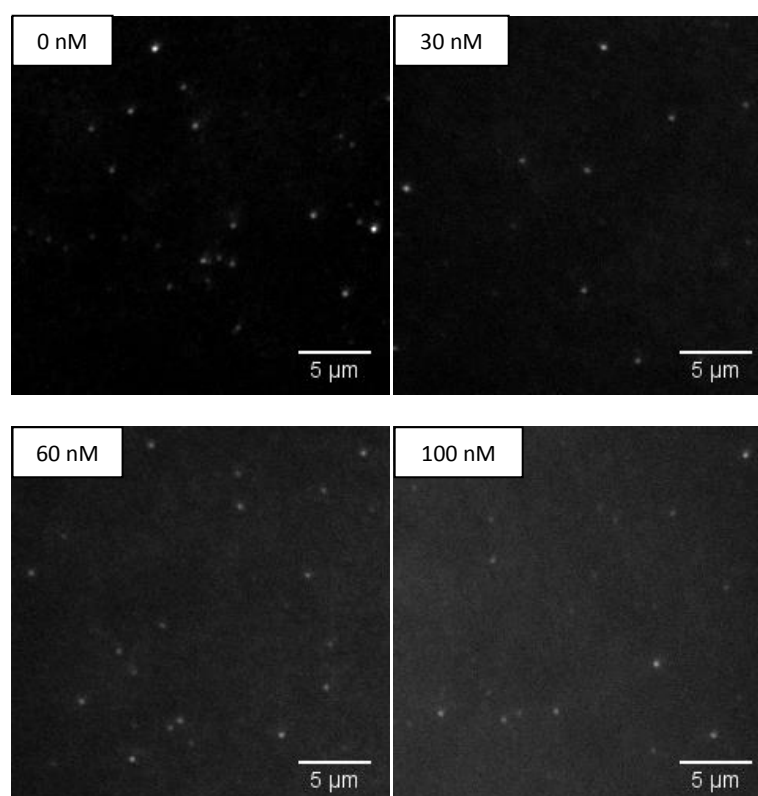


Figure 6.18. Background fluorescence from Cy3-ATP in solution.

Cy3-labelled ChID subunits were immobilized and imaged in chelatase buffer (50 mM MOPS/KOH, 0.3 M glycerol, pH7.7 at 25 °C) containing gloxy and varying concentrations of Cy3-ATP (indicated in the top left of each image) with TIR illumination at 514 nm and a 10 ms exposure.

The restriction on the maximum concentration of fluorescent nucleotide required further concessions to be made in the experimental design. At 60 nM Cy3-ATP, out of approximately 150 protein molecules present in a field of view (using protein concentrations as previously), only 6 could be expected to bind fluorescent nucleotide at any one time. This was too few fluorophores to be easily discerned against an elevated background fluorescence, so the protein concentration was increased. Increasing the protein concentration meant that labelled protein could not be used since the complexes were too closely spaced to be distinguished. As a consequence, unlabelled His₆-tagged ChII was used. Very small quantities of Cy3-labelled ChID subunits (approximately 2 to 3 molecules per field of view) were included in the samples to facilitate focussing of the microscope.

Analysis of recorded image series was performed as detailed in Section 4.3. Briefly, fluorophores were identified and localized, and their intensity was extracted from every frame of the image series after background removal to create intensity-time traces. Each intensity trace was subjected to half-amplitude threshold analysis, where a fluorescent nucleotide is deemed to be bound as long as the intensity is greater than half the maximum signal intensity. Bound or dwell times were then extracted, and individual histograms of dwell times were constructed for each identified fluorescent spot. Since the protein had to be immobilized at high concentration to compensate for the low concentration of

nucleotide, it is possible that the bound times observed for one spot correspond to several individual protein molecules. Given the low nucleotide concentration and binding affinity for the protein, it is unlikely that more than one protein molecule in a diffraction-limited area bound nucleotide at any one time, so observed bound times will generally correspond to single non-temporally overlapping events. Single and double exponential distributions were fitted to each histogram using maximum likelihood estimation, yielding average lifetimes for the bound events of each fluorescent spot.

Two different variables were investigated in visualizing Cy3-ATP binding to immobilized ChII: the location of the His tag (N- or C-terminal to the protein), and the salt content of the buffer (normal chelatase buffer and high salt chelatase buffer). The high salt buffer (50 mM MOPS/KOH, 500 mM KCl, 0.3 M glycerol, pH7.7 at 25 °C) was expected to shield the protein from the functionalized surface and disrupt any electrostatic interactions which might be deforming the protein and disrupting nucleotide binding. A blank control prepared in the same manner as the sample, but omitting the addition of protein, was subjected to the same image analysis and compared to each sample.

Figures 6.19 to 6.22 show the distributions of the two time components (average lifetimes) extracted from double exponential fits to bound times for the different conditions. Each average lifetime distribution is displayed alongside the corresponding distribution for the control. Unfortunately, binding-like events were detected in the control, making it more difficult to uniquely identify real binding events in the sample. These might have been due to fluorophores intrinsic to the functionalized coverslip exhibiting blinking behaviour, or free nucleotide adhering to the surface. The relatively flexible EDA linker attaching the Cy3 fluorophore to the nucleotide could allow the fluorophore to explore a limited spatial volume and produce binding-like behaviour during interactions with the surface. It is unlikely these were caused by freely diffusing Cy3-ATP molecules entering and leaving the evanescent illumination field since the diffusion rate of Cy3-ATP in aqueous buffer at room temperature is approximately $2 \times 10^{-6} \text{ cm}^2 \cdot \text{s}^{-1}$ [95,128]. In one frame (20ms) its root mean displacement is about 4 μm or 8 PSFs. However, more events were consistently detected in the sample than the control, indicating that real binding events were also being detected. The location of the His tag did not affect the accessibility of the nucleotide binding site as both N- and C-terminally tagged protein samples exhibited a greater number of events than the control. The C-terminally tagged protein samples exhibited slightly longer bound lifetimes than the N-terminally tagged protein samples in both buffers, suggesting that the nucleotide binding site is less accessible in protein immobilized through a His tag on the C terminus.

The distribution of the shorter average lifetime (panels A and B in each figure) centres around 30 ms in both the samples and the controls. This most probably reflects the time resolution of the system since the data were recorded at 20 ms per frame and a resolution limit of 54 ms was imposed on the data during processing. It is possible that binding events with lifetimes below the resolution time of the system have been missed, particularly since the shorter time component from exponential fitting is of the order of the system temporal resolution. If several of these events occur consecutively they could become averaged through one frame length and appear as longer events with smaller amplitudes. This would

skew the lifetime distributions to longer times, causing an underestimate of the associated rate constant.

The distribution of the longer lifetime (panels C and D in each figure) remains relatively constant in the controls with a maximum between 227 and 240 ms, but slight differences can be seen between the samples. Fitting the sample distributions to a mixture of two Gaussians produces one peak with a mean similar to that of the control distribution, and another with mean corresponding to the sample signal. The distributions for the samples in normal chelatase buffer (panel C Figures 6.19 and 6.20) show peaks at 161 ms and 171 ms, suggesting an off-rate constant of $6.0 \pm 1.9 \text{ s}^{-1}$. (The off-rate constant is the inverse of the binding lifetime). This shifts to slightly longer times, 321 ms and 387 ms, for the samples in the high salt buffer ($2.8 \pm 1.2 \text{ s}^{-1}$). The higher salt concentration is noticeably affecting the nucleotide binding kinetics. Salt concentration is known to affect the oligomerization of members of the AAA⁺ superfamily[129–131] (increasing ionic strength causes a decrease in the size of oligomer formed), and as such kinetic investigations of magnesium chelatase have controlled the ionic strength of the buffer at $I = 0.1$ [8,12].

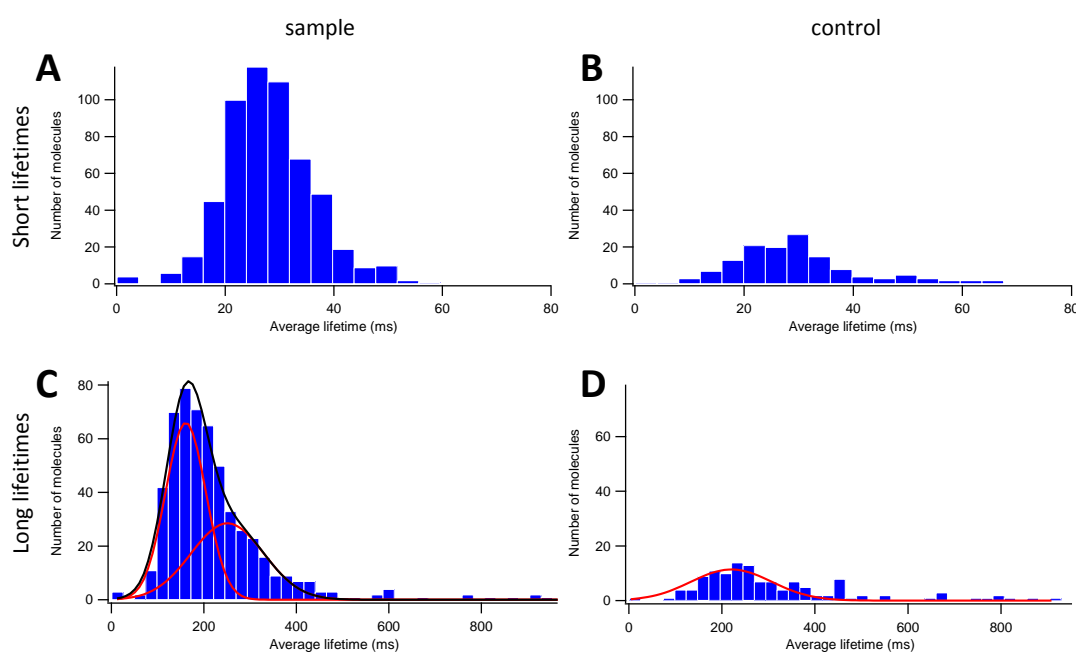


Figure 6.19. Average lifetime distributions for N-terminally tagged Chl in normal chelatase buffer.

Buffer contained 10 mM free Mg^{2+} and 60 nM Cy3-ATP. Double exponential fits to bound time distributions for samples and controls yielded two time components. The distribution of short lifetimes is shown for A) the sample and B) the control. The distribution of long lifetimes is shown for C) the sample with Gaussian fits giving means (\pm SD) of 161 ± 44 ms and 251 ± 79 ms, and D) the control with mean 240 ± 91 ms.

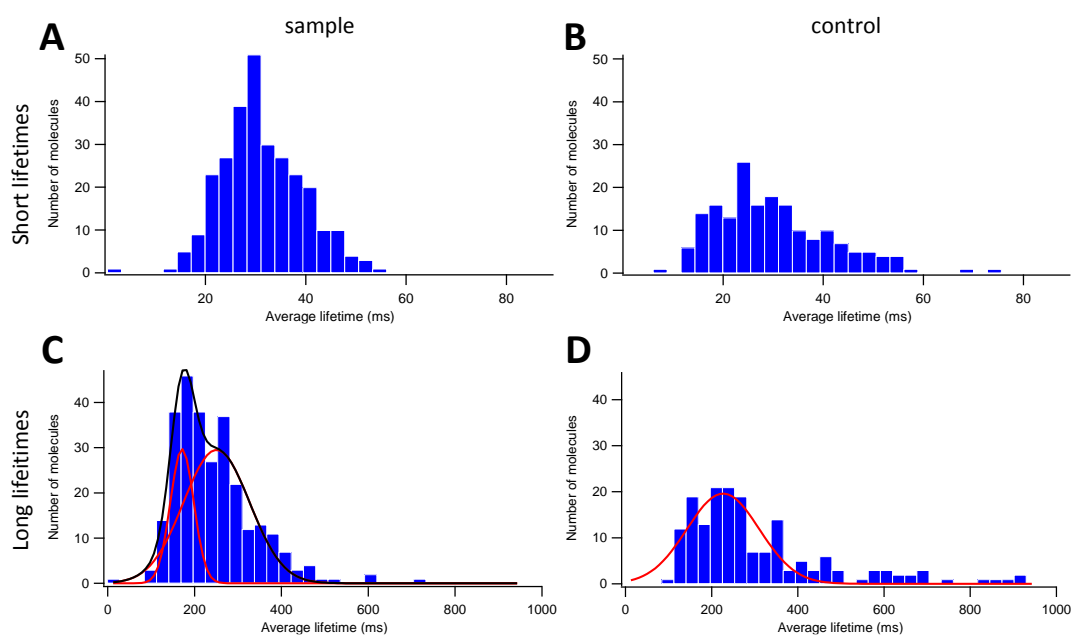


Figure 6.20. Average lifetime distributions for C-terminally tagged ChII in normal chelatase buffer.

Buffer contained 10 mM free Mg^{2+} and 60 nM Cy3-ATP. Double exponential fits to bound time distributions for samples and controls yielded two time components. The distribution of short lifetimes is shown for A) the sample and B) the control. The distribution of long lifetimes is shown for C) the sample with Gaussian fits giving means (\pm SD) of 171 ± 27 ms and 251 ± 76 ms, and D) the control with mean 227 ± 84 ms.

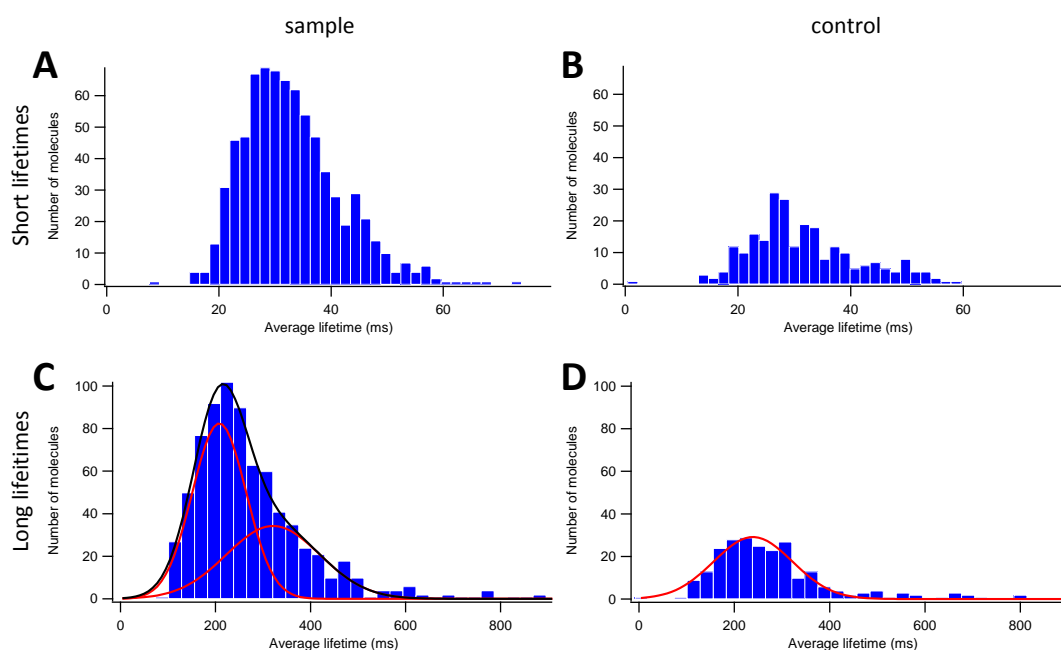


Figure 6.21. Average lifetime distributions for N-terminally tagged ChII in high salt chelatase buffer.

Buffer contained 10 mM free Mg^{2+} and 60 nM Cy3-ATP. Double exponential fits to bound time distributions for samples and controls yielded two time components. The distribution of short lifetimes is shown for A) the sample and B) the control. The distribution of long lifetimes is shown for C) the sample with Gaussian fits giving means (\pm SD) of 208 ± 57 ms and 321 ± 98 ms, and D) the control with mean 239 ± 82 ms.

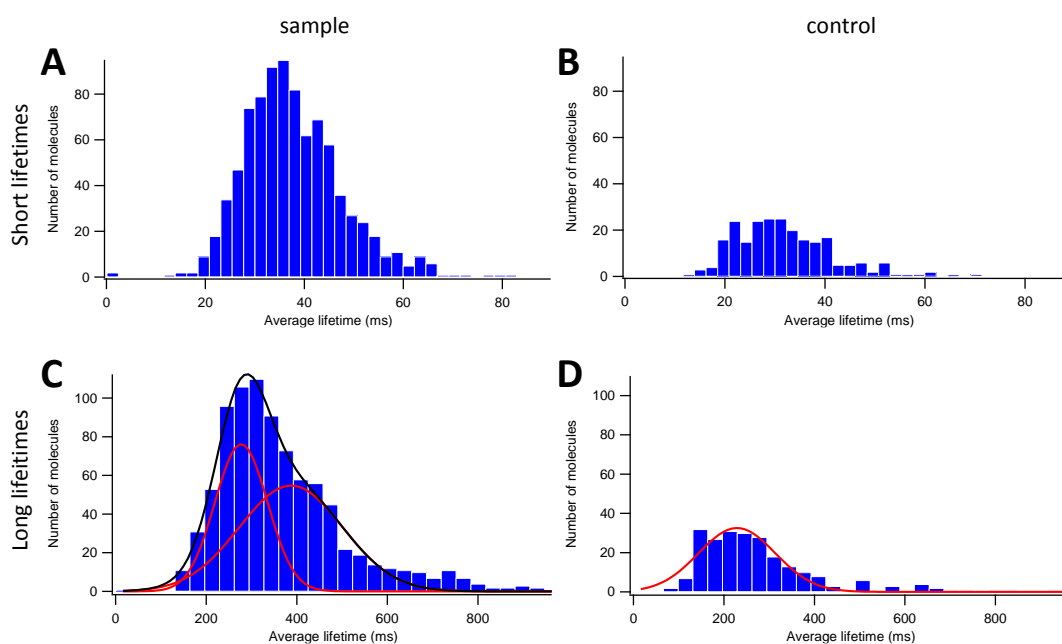
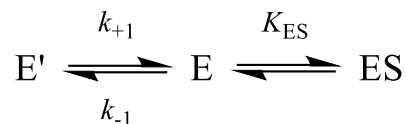


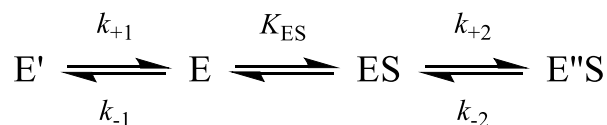
Figure 6.22. Average lifetime distributions for C-terminally tagged ChII in high salt chelatase buffer.

Buffer contained 10 mM free Mg^{2+} and 60 nM Cy3-ATP. Double exponential fits to bound time distributions for samples and controls yielded two time components. The distribution of short lifetimes is shown for A) the sample and B) the control. The distribution of long lifetimes is shown for C) the sample with Gaussian fits giving means (\pm SD) of 277 ± 58 ms and 387 ± 114 ms, and D) the control with mean 229 ± 85 ms.

A previous transient kinetics study of TNP-ATP binding to ChII proposed the model shown in Scheme 6.1 in which the free enzyme isomerizes before binding substrate[12] (the conformational selection mechanism of substrate binding, in contrast to induced fit where substrate binding is followed by an isomerization of the enzyme to attain the optimal binding conformation). It was assumed that the substrate binding step was fast compared to the isomerization (the rapid equilibrium approximation) to describe stopped flow measurements in which the observed rate constant decreased with increasing TNP-ATP concentration. At 10 mM free Mg^{2+} (the same buffer conditions as used for the current normal chelatase buffer samples) the rate constants found for the isomerization step were $k_{+1} = 3.2 \pm 0.3 \text{ s}^{-1}$ and $k_{-1} = 16.9 \pm 51.5 \text{ s}^{-1}$. This previous experimental arrangement was able to probe the behaviour of the free enzyme species using the change in fluorescence of TNP-ATP upon binding to protein. The current work investigates the substrate binding step directly through observing the bound times of Cy3-ATP, and so should be probing the off-rate constant of the E to ES step in Scheme 6.1. If the temporal resolution of the experimental set-up has not artificially reduced the measured off-rate constant, the value of 6.0 s^{-1} (comparable in magnitude to the isomerization rate constants) found for the dissociation of nucleotide from ChII in normal chelatase buffer suggests that Scheme 6.1 is too simple in the rapid equilibrium case. Maintaining the assumption that K_{ES} is fast compared to the previously postulated free enzyme isomerization, this discrepancy could be explained by the enzyme undergoing a further slow isomerization while binding substrate (Scheme 6.2). The rate constant accessed by single molecule studies is then k_{-2} .



Scheme 6.1. Isomerization of enzyme before substrate binding.



Scheme 6.2. Isomerization of enzyme before and after binding substrate.

It is expected that a further slow isomerization step would produce stopped flow traces with a double exponential character. This was not found by Reid et al[12] however it was noted in the study that approximately 60 % of the total fluorescence change from nucleotide binding occurred within the dead time of the instrument, and it was not possible to monitor binding at nucleotide concentrations below 5 μ M due to the weak fluorescence emission of TNP-ATP even when bound. The resulting noisy traces might hide a second exponential component.

Alternatively, a more recent review of the conformational selection/induced fit dichotomy proposed that in the more general case when rapid equilibrium is not used as a simplifying assumption, the expression for the smaller observed rate constant k_{obs} derived from the conformational selection model (Equation 6.3) is capable of producing both an inversely proportional and a proportional relationship between k_{obs} and the ligand concentration[132]. That is, the expression in Equation 6.3 produces the inverse relationship between k_{obs} and $[S]$ when $k_{+1} < k_{off} < k_{-1} + k_{+1}$. In this situation, the off-rate constant for nucleotide binding takes on values comparable in magnitude to the isomerization rate constants.

$$k_{obs} = \frac{k_{-1} + k_{+1} + k_{off} + k_{on}[S] - \sqrt{(k_{off} + k_{on}[S] - k_{-1} - k_{+1})^2 + 4k_{-1}k_{on}[S]}}{2}$$

Equation 6.3. General expression for the smaller observed rate constant in the case of conformational selection

The current study has used single molecule imaging to measure dwell times of fluorescent nucleotide binding to the ATPase subunit of magnesium chelatase. Initial attempts were made with TNP-ATP which had been shown previously to bind to ChII, and to be displaced by the natural substrate ATP, indicating that it was accessing the nucleotide binding pocket. The fluorescence emission of TNP-ATP is low in aqueous solution and increases upon binding to protein, making this an attractive candidate for single molecule substrate binding studies where the background fluorescence from free nucleotide often limits the concentration which can be used. The fluorescence emission of bound TNP-ATP was too low however to be detected against the high background fluorescence typical of glass substrates under the shorter wavelengths of the visible spectrum.

Another fluorescent nucleotide was therefore chosen with a longer excitation wavelength, Cy3-ATP. Cy3-ATP was found to bind to ChII through competition with TNP-ATP, indicating that it also is an alternative ligand for this protein. Despite competing signal from free fluorescent nucleotide interacting with the functionalized surface, samples exhibited a time constant unique from any seen in controls. The observation of a short time component of magnitude similar to the system temporal resolution suggests that short events might have been missed or counted as longer, smaller amplitude events. However, the rate constant yielded by the current work for the release of nucleotide from ChII is not incompatible with past experiments. It implies that the binding model inferred from the rapid equilibrium assumption used in the interpretation of previous results was either too simplistic (requiring the addition of a further isomerization step after nucleotide binding), or that the use of the rapid equilibrium approximation led to incorrect conclusions about the values of the rate constants. The rate constants associated with nucleotide binding could be of a similar order of magnitude to those of the isomerization step. While it is clear that the kinetic mechanism of binding for ChII is conformational selection, further stopped flow studies using different ligands with varying K_d would determine the existence of the proposed second isomerization step.

7. Summary

Magnesium chelatase catalyses the insertion of magnesium into protoporphyrin IX to form magnesium protoporphyrin IX in the first committed step of chlorophyll biosynthesis. It comprises three different subunits known as ChII, ChID and ChIH. ChII and ChID interact in an ATP-dependent manner, forming a complex to which the porphyrin-carrying subunit ChIH binds[11,19,133]. Magnesium insertion requires a large amount of energy provided by the concomitant ATP hydrolysis of the ChII subunit[29].

The questions of the exact structure formed by the active ID complex and the ATP binding kinetics of the AAA⁺ subunit ChII have long lain unanswered, though not from lack of investigation[14,17–19,25,37]. Single molecule fluorescence microscopy was employed as a new way to dissect the complex workings of this enzyme. In contrast to bulk solution techniques such as titration experiments, equilibrium binding, and stopped flow measurements, single molecule techniques access the structure of protein complexes at the subunit level, and binding dynamics between individual protein molecules and their ligands. This enables heterogeneities within a population to be seen which would be averaged out by bulk techniques[92,93].

The work presented outlines the challenges faced in adapting a relatively new experimental technique to address questions which have to date eluded definitive answers from standard biochemical assays. While experimental limitations impacted the quantity of information which could be extracted from the data, it was still possible to gain new insight. Single molecule subunit counting assays shed light on the dynamic nature of the ID complex while nucleotide binding experiments suggest conclusions drawn from previous transient kinetic studies require revision.

7.1. Counting ChID subunits in the ID complex

In order to investigate the structure of the ID complex via single molecule stepwise photobleaching, the ChII and ChID subunits were fluorescently labelled. Maleimide chemistry was chosen as a simple, specific way to label the subunits which contain naturally occurring thiols within their cysteine residues. Upon labelling, ChII was found to lose all chelatase activity while that of labelled ChID subunits was reduced to approximately 30 % of normal. Both labelled subunits were still capable of interacting to form an ID complex. ChID was therefore chosen to be the labelled subunit in subunit counting assays as it could still form an active complex.

Initial analysis of label number distributions extracted from stepwise photobleaching traces of immobilized complexes suggested that fluorescent labels were being detected with a probability of around 0.5, which is unexpectedly low for an organic dye[68,73]. Modelling of the data based on the observed fluorophore lifetime and SNR revealed a higher label detection probability and a very different label number distribution (Figure 5.14), suggesting that the assumption of a homogeneous complex distribution used in the binomial fits to the experimental label number distributions was incorrect. It was concluded that the underlying distribution of ID complexes contained varying numbers of ChID subunits ranging from 1 to 5 (Figure 7.1).

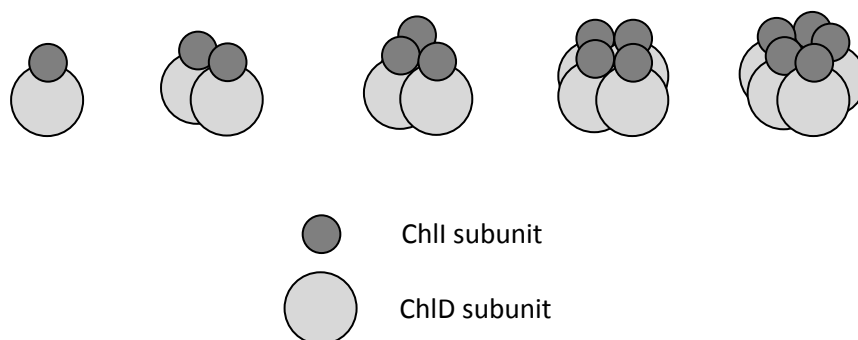


Figure 7.1. Possible ID complex stoichiometries present in single molecule photobleaching experiments.

Single molecule stepwise photobleaching analysis suggests ID complexes contain varying numbers of ChID subunits ranging from 1 to 5. Previous titration experiments and pull down assays between the I and D subunits suggest that ChII and ChID form complexes in a 1:1 ratio, as illustrated above. It is possible that the number of I subunits associating in a complex is not strictly 1:1.

It was not possible to extract information on the relative proportions of complexes containing a certain number of subunits nor was it possible to say whether all of the potential subunit configurations represented an active complex. This limitation is a feature of the subunit counting technique[84]. The label number distribution cannot be directly interpreted as a subunit number distribution due to the non-zero probability of missing a photobleaching event. As such, single molecule subunit counting has limited applicability to systems where the complex population is heterogeneous. It can be used to estimate a lower limit for the maximum complex size (the highest observed label number), and to determine that a mixture of complexes exists by comparison of observed label number distributions with expected binomial distributions. However it is not possible to draw firm conclusions on the distribution of a heterogeneous population of complexes without fluorescent labels for which the labelling efficiency is 1:1 and the detection efficiency approaches 100 %.

7.2. Single molecule nucleotide binding to ChII

Single molecule studies of nucleotide binding to ChII required the use of a suitable fluorescent nucleotide analogue. TNP-ATP was chosen on the basis that it was already known to bind ChII and it undergoes a significant fluorescence enhancement upon binding to the protein[12]. It was shown to sustain the ID complex, but acted as a mixed inhibitor to the chelatase reaction suggesting it binds elsewhere on the chelatase complex as well as at the nucleotide binding site of ChII. Attempts to visualize binding of TNP-ATP to the ID complex via fluorescence microscopy were frustrated by the high fluorescence background emitted from the coverslip substrate when illuminated by the short wavelength excitation light required for TNP-ATP. It is also possible that the enhancement in the fluorescence emission of TNP-ATP is not sufficient to be detected at the short exposure times required to capture nucleotide binding dynamics.

The fluorescent nucleotide Cy3-ATP was then chosen for its proven photostability and reasonably high quantum yield[92,126]. Cy3-ATP was shown to be an alternative ligand for

ChII through a competition binding experiment with TNP-ATP. In this competition experiment, Cy3-ATP was found to have a steady state dissociation constant on the order of 1 μ M. In single molecule nucleotide binding experiments, controls run under the same conditions as samples exhibited binding-like events attributable either to fluorophores within the coverslip substrate or non-specific binding of the nucleotide to the coverslip. More events were consistently seen in samples compared to controls, indicating that real binding events were being observed. Bound times were extracted from intensity traces by half-amplitude thresholding and the resulting distributions were fitted by Maximum Likelihood Estimation to exponential decays. It was found that both samples and controls displayed two time components. The short time component in both the sample and control was consistent with the temporal resolution of the system and data analysis. The distribution of the long time component for the control had a mean of approximately 230 ms. The long time component distribution for the sample was fitted with two overlapping Gaussian distributions, one of which had a mean value close to that of the control. The other mean was taken as being the time constant for real binding events which produced an off-rate constant for Cy3-ATP binding to ChII of $(6.0 \pm 1.9) \text{ s}^{-1}$.

7.3. Conclusions and future work

The number of ChID subunits associating into an ID complex is variable, ranging from 1 to 5 while the most consistent I:D stoichiometry emerging from previous titration experiments is 1:1 or 2:1 [17,25]. Combining these two results suggests that ID complexes exist in a heterogeneous mix ranging from heterodimers (1 ChII and 1 ChID) to double pentamers. It would be useful to count the numbers of ChII subunits directly to determine the likely range of the subunit distribution. The stoichiometry of the complex could be estimated from a comparison of the label number distributions for each subunit, or determined directly using a different label for each of the two subunits. This would also allow a comparison to be made with the stoichiometry suggested by titration experiments. This would require either a suitable label which does not inactivate the ChII subunit, or the use of mutants.

The off-rate constant found for nucleotide binding to ChII is of the same order of magnitude as the isomerization rate constants found previously by stopped flow experiments[12]. This suggests that the reaction scheme proposed to explain the previous results requires a further isomerization step to be added after the enzyme binds nucleotide (Schemes 6.1 and 6.2). The nucleotide-bound isomerization is then the slower rate constant which is observed by single molecule binding experiments. Alternatively, it might not be appropriate to use a rapid equilibrium approximation in interpreting transient kinetic data since it is possible to have binding rate constants of the same order of magnitude as isomerization rate constants and still observe an inversely proportional relationship between binding signal and nucleotide concentration (Equation 6.3)[132]. Further work is needed to resolve this question. It is not recommended that single molecule binding studies are pursued further as the temporal resolution limit of the system was evident in the data. Further stopped flow studies using different ligands with varying K_d would determine the existence of the proposed second isomerization step.

References

- 1 Blankenship, R. E. (2014) *Molecular Mechanisms of Photosynthesis* 2nd editio., Wiley Blackwell, Chichester.
- 2 Scheer, H. (2006) An Overview of Chlorophylls and Bacteriochlorophylls: Biochemistry, Biophysics, Functions and Applications. In *Chlorophylls and Bacteriochlorophylls: Biochemistry, Biophysics, Functions and Applications* (Advances in Photosynthesis and Respiration Vol. 25) (Grimm, B., Porra, R. J., Rüdiger, W., and Scheer, H., eds.), pp 1–26, Springer, Dordrecht.
- 3 Yaronskaya, E. and Grimm, B. (2006) The Pathway from 5-Aminolevulinic Acid to Protochlorophyllide and Protoheme. In *Chlorophylls and Bacteriochlorophylls: Biochemistry, Biophysics, Functions and Applications* (Advances in Photosynthesis and Respiration Vol. 25) (Grimm, B., Porra, R. J., Rüdiger, W., and Scheer, H., eds.), pp 173–188, Springer, Dordrecht.
- 4 Rüdiger, W. and Grimm, B. (2006) Chlorophyll Metabolism, an Overview. In *Chlorophylls and Bacteriochlorophylls: Biochemistry, Biophysics, Functions and Applications* (Advances in Photosynthesis and Respiration Vol. 25) (Grimm, B., Porra, R. J., Rüdiger, W., and Scheer, H., eds.), pp 133–146, Springer, Dordrecht.
- 5 Gorchein, A. (1972) Magnesium protoporphyrin chelatase activity in *Rhodospseudomonas spheroides*. Studies with whole cells. *Biochem. J.* **127**, 97–106.
- 6 Gibson, L. C., Willows, R. D., Kannangara, C. G., von Wettstein, D. and Hunter, C. N. (1995) Magnesium-Protoporphyrin Chelatase of *Rhodobacter sphaeroides*: Reconstitution of Activity by Combining the Products of the *bchH*, *-I*, and *-D* Genes Expressed in *Escherichia coli*. *Proc. Natl. Acad. Sci. U. S. A.* **92**, 1941–1944.
- 7 Jones, O. T. G. (1978) Biosynthesis of porphyrins, hemes and chlorophylls. In *The photosynthetic bacteria* (Clayton, R. K., and Sistrom, W. R., eds.), pp 751–777, Plenum Press, New York.
- 8 Viney, J., Davison, P. A., Hunter, C. N. and Reid, J. D. (2007) Direct measurement of metal-ion chelation in the active site of the AAA+ ATPase magnesium chelatase. *Biochemistry* **46**, 12788–94.
- 9 Jensen, P. E., Gibson, L. E., Henningsen, K. W. and Hunter, C. N. (1996) Expression of the *chlI*, *chlD*, and *chlH* Genes from the Cyanobacterium *Synechocystis* PCC6803 in *Escherichia coli* and Demonstration That the Three Cognate Proteins Are Required for Magnesium-protoporphyrin Chelatase Activity. *J. Biol. Chem.* **271**, 16662–16667.
- 10 Fodje, M. N., Hansson, A., Hansson, M., Olsen, J. G., Gough, S., Willows, R. D. and Al-Karadaghi, S. (2001) Interplay between an AAA module and an integrin I domain may regulate the function of magnesium chelatase. *J. Mol. Biol.* **311**, 111–22.
- 11 Jensen, P. E., Gibson, L. C. and Hunter, C. N. (1999) ATPase activity associated with the magnesium-protoporphyrin IX chelatase enzyme of *Synechocystis* PCC6803: evidence for ATP hydrolysis during Mg²⁺ insertion, and the MgATP-dependent interaction of the *ChlI* and *ChlD* subunits. *Biochem. J.* **339** (Pt 1), 127–34.
- 12 Reid, J. D., Siebert, C. A., Bullough, P. A. and Hunter, C. N. (2003) The ATPase activity of the *ChlI* subunit of magnesium chelatase and formation of a heptameric AAA+ ring. *Biochemistry* **42**, 6912–20.
- 13 Neuwald, A. F., Aravind, L., Spouge, J. L. and Koonin, E. V. (1999) AAA+: A Class of Chaperone-Like ATPases Associated with the Assembly, Operation, and Disassembly of Protein Complexes. *Genome Res.* **9**, 27–43.

- 14 Lundqvist, J., Elmlund, H., Wulff, R. P., Berglund, L., Elmlund, D., Emanuelsson, C., Hebert, H., Willows, R. D., Hansson, M., Lindahl, M., et al. (2010) ATP-induced conformational dynamics in the AAA+ motor unit of magnesium chelatase. *Structure* **18**, 354–65.
- 15 Lake, V., Olsson, U., Willows, R. D. and Hansson, M. (2004) ATPase activity of magnesium chelatase subunit I is required to maintain subunit D in vivo. *Eur. J. Biochem.* **271**, 2182–2188.
- 16 Willows, R. D., Hansson, A., Birch, D., Al-Karadaghi, S. and Hansson, M. (2004) EM single particle analysis of the ATP-dependent Bchl complex of magnesium chelatase: an AAA+ hexamer. *J. Struct. Biol.* **146**, 227–33.
- 17 Jensen, P. E., Gibson, L. C. and Hunter, C. N. (1998) Determinants of catalytic activity with the use of purified I, D and H subunits of the magnesium protoporphyrin IX chelatase from *Synechocystis* PCC6803. *Biochem. J.* **334**, 335–44.
- 18 Willows, R. D. and Beale, S. I. (1998) Heterologous Expression of the *Rhodobacter capsulatus* Bchl, -D, and -H Genes That Encode Magnesium Chelatase Subunits and Characterization of the Reconstituted Enzyme. *J. Biol. Chem.* **273**, 34206–34213.
- 19 Gibson, L. C., Jensen, P. E. and Hunter, C. N. (1999) Magnesium chelatase from *Rhodobacter sphaeroides*: initial characterization of the enzyme using purified subunits and evidence for a Bchl-BchD complex. *Biochem. J.* **337**, 243–51.
- 20 Hanson, P. I. and Whiteheart, S. W. (2005) AAA+ proteins: have engine, will work. *Nat. Rev. Mol. Cell Biol.* **6**, 519–29.
- 21 Petersen, B. L., Kannangara, C. G. and Henningsen, K. W. (1999) Distribution of ATPase and ATP-to-ADP phosphate exchange activities in magnesium chelatase subunits of *Chlorobium vibrioforme* and *Synechocystis* PCC6803. *Arch. Microbiol.* **171**, 146–150.
- 22 Adams, N. B. P. and Reid, J. D. (2012) Nonequilibrium isotope exchange reveals a catalytically significant enzyme-phosphate complex in the ATP hydrolysis pathway of the AAA(+) ATPase magnesium chelatase. *Biochemistry* **51**, 2029–31.
- 23 Hansson, M. and Kannangara, C. G. (1997) ATPases and phosphate exchange activities in magnesium chelatase subunits of *sphaeroides*. *Proc. Natl. Acad. Sci. U. S. A.* **94**, 13351–13356.
- 24 Vale, R. D. (2000) AAA Proteins: Lords of the Ring. *J. Cell Biol.* **150**, F13–F19.
- 25 Adams, N. B. P. (2012) The roles of ChII and ChID, the AAA+ subunits of Magnesium Chelatase., The University of Sheffield.
- 26 Lundqvist, J., Braumann, I., Kurowska, M., Müller, A. H. and Hansson, M. (2013) Catalytic turnover triggers exchange of subunits of the magnesium chelatase AAA+ motor unit. *J. Biol. Chem.* **288**, 24012–9.
- 27 Sirijovski, N., Olsson, U., Lundqvist, J., Al-Karadaghi, S., Willows, R. D. and Hansson, M. (2006) ATPase activity associated with the magnesium chelatase H-subunit of the chlorophyll biosynthetic pathway is an artefact. *Biochem. J.* **400**, 477–84.
- 28 Sirijovski, N., Lundqvist, J., Rosenbäck, M., Elmlund, H., Al-Karadaghi, S., Willows, R. D. and Hansson, M. (2008) Substrate-binding model of the chlorophyll biosynthetic magnesium chelatase BchH subunit. *J. Biol. Chem.* **283**, 11652–60.
- 29 Reid, J. D. and Hunter, C. N. (2004) Magnesium-dependent ATPase Activity and Cooperativity of Magnesium Chelatase from *Synechocystis* sp. PCC6803. *J. Biol. Chem.* **279**, 26893–26899.

- 30 Adams, N. B. P. and Reid, J. D. (2013) The allosteric role of the AAA domain of ChlD protein from the magnesium chelatase of *synechocystis* species PCC 6803. *J. Biol. Chem.* **288**, 28727–28732.
- 31 Karger, G. A., Reid, J. D. and Hunter, C. N. (2001) Characterization of the Binding of Deuteroporphyrin IX to the Magnesium Chelatase H Subunit and Spectroscopic Properties of the Complex †. *Biochemistry, American Chemical Society* **40**, 9291–9299.
- 32 Jensen, P., Reid, J. and Hunter, C. (2000) Modification of cysteine residues in the ChlI and ChlH subunits of magnesium chelatase results in enzyme inactivation. *Biochem. J.* **352**, 435–441.
- 33 Amor-Mahjoub, M., Suppini, J.-P., Gomez-Vrielyunck, N. and Ladjimi, M. (2006) The effect of the hexahistidine-tag in the oligomerization of HSC70 constructs. *J. Chromatogr. B, Anal. Technol. Biomed. life Sci.* **844**, 328–34.
- 34 Wu, J. and Filutowicz, M. (1999) Hexahistidine (His6)-tag dependent protein dimerization: A cautionary tale. *Acta Biochim. Pol.* **46**, 591–599.
- 35 Petersen, B. L., Jensen, P. E., Gibson, L. C. D., Stummann, B. M., Hunter, C. N. and Henningsen, K. W. (1998) Reconstitution of an active magnesium chelatase enzyme complex from the bchl, -D, and -H gene products of the green sulfur bacterium *Chlorobium vibrioforme* expressed in *Escherichia coli*. *J. Bacteriol.* **180**, 699–704.
- 36 Walker, C. J. and Weinstein, J. D. (1994) The magnesium-insertion step of chlorophyll biosynthesis is a two-stage reaction. *Biochem. J.* **299**, 277–84.
- 37 Elmlund, H., Lundqvist, J., Al-Karadaghi, S., Hansson, M., Hebert, H. and Lindahl, M. (2008) A new cryo-EM single-particle ab initio reconstruction method visualizes secondary structure elements in an ATP-fueled AAA+ motor. *J. Mol. Biol.* **375**, 934–47.
- 38 Qian, P., Marklew, C. J., Viney, J., Davison, P. A., Brindley, A. A., Söderberg, C., Al-Karadaghi, S., Bullough, P. A., Grossmann, J. G. and Hunter, C. N. (2012) Structure of the cyanobacterial Magnesium Chelatase H subunit determined by single particle reconstruction and small-angle X-ray scattering. *J. Biol. Chem.* **287**, 4946–56.
- 39 Willows, R. D. (2003) Biosynthesis of chlorophylls from protoporphyrin IX. *Nat. Prod. Rep.* **20**, 327.
- 40 Funatsu, T., Harada, Y., Tokunaga, M., Saito, K. and Yanagida, T. (1995) Imaging of single fluorescent molecules and individual ATP turnovers by single myosin molecules in aqueous solution. *Nature* **374**, 555–559.
- 41 Kubitscheck, U., Kückmann, O., Kues, T. and Peters, R. (2000) Imaging and Tracking of Single GFP Molecules in Solution. *Biophys. J.* **78**, 2170–2179.
- 42 Schütz, G. J., Schindler, H. and Schmidt, T. (1997) Single-molecule microscopy on model membranes reveals anomalous diffusion. *Biophys. J.* **73**, 1073–80.
- 43 Zhang, H. and Guo, P. (2014) Single molecule photobleaching (SMPB) technology for counting of RNA, DNA, protein and other molecules in nanoparticles and biological complexes by TIRF instrumentation. *Methods, Elsevier Inc.* **67**, 169–76.
- 44 Huang, B., Bates, M. and Zhuang, X. (2009) Super-resolution fluorescence microscopy. *Annu. Rev. Biochem.* **78**, 993–1016.
- 45 Ye, J. Y., Yamane, Y., Yamauchi, M., Nakatsuka, H. and Ishikawa, M. (2000) Direct observation of the interaction of single fluorescent nucleotide analogue molecules with DNA polymerase I. *Chem. Phys. Lett.* **320**, 607–612.

- 46 Ho, P. S., Mergny, J.-L., Krüger, A. C. and Birkedal, V. (2013) Single molecule FRET data analysis procedures for FRET efficiency determination: Probing the conformations of nucleic acid structures. *Methods* **64**, 36–42.
- 47 Herman, B. (1998) *Fluorescence Microscopy* (Royal Microscopical Society Microscopy Handbooks 40) 2nd Editio., BIOS Scientific Publishers, Oxford.
- 48 Lakowicz, J. R. (2006) Introduction to Fluorescence. In *Principles of Fluorescence Spectroscopy* 3rd ed., pp 1–26, Springer.
- 49 Turro, N. J., Ramamurthy, V. and Scaiano, J. C. (2009) *Principles of Molecular Photochemistry: An Introduction* (Stiefel, J., ed.), University Science Books, Sausalito, California.
- 50 Hecht, E. (2002) *Optics* 4th Intern., Addison Wesley, San Francisco.
- 51 Wang, L., Vasilev, C., Canniffe, D. P., Wilson, L. R., Hunter, C. N. and Cadby, A. J. (2012) Highly confined surface imaging by solid immersion total internal reflection fluorescence microscopy. *Opt. Express* **20**, 3311.
- 52 Axelrod, D. (2001) Total internal reflection fluorescence microscopy in cell biology. *Traffic* **2**, 764–774.
- 53 Bobroff, N. (1986) Position measurement with a resolution and noise-limited instrument. *Rev. Sci. Instrum.* **57**, 1152–1157.
- 54 Thompson, R. E., Larson, D. R. and Webb, W. W. (2002) Precise nanometer localization analysis for individual fluorescent probes. *Biophys. J.* **82**, 2775–83.
- 55 Cheezum, M. K., Walker, W. F. and Guilford, W. H. (2001) Quantitative Comparison of Algorithms for Tracking Single Fluorescent Particles. *Biophys. J.*, Elsevier **81**, 2378–2388.
- 56 Lee, G. M., Ishihara, A. and Jacobson, K. A. (1991) Direct observation of brownian motion of lipids in a membrane. *Proc. Natl. Acad. Sci. U. S. A.* **88**, 6274–8.
- 57 Ghosh, R. N. and Webb, W. W. (1994) Automated detection and tracking of individual and clustered cell surface low density lipoprotein receptor molecules. *Biophys. J.* **66**, 1301–18.
- 58 Gelles, J., Schnapp, B. J. and Sheetz, M. P. (1988) Tracking kinesin-driven movements with nanometre-scale precision. *Nature* **331**, 450–453.
- 59 Shin, Y., Du, Y., Collier, S. E., Ohi, M. D., Lang, M. J. and Ohi, R. (2015) Biased Brownian motion as a mechanism to facilitate nanometer-scale exploration of the microtubule plus end by a kinesin-8. *Proc. Natl. Acad. Sci.* **112**, E3826–E3835.
- 60 Stallinga, S. and Rieger, B. (2010) Accuracy of the gaussian point spread function model in 2D localization microscopy. *Opt. Express, Optical Society of America* **18**, 24461–76.
- 61 Zhang, B., Zerubia, J. and Olivo-Marin, J.-C. (2007) Gaussian approximations of fluorescence microscope point-spread function models. *Appl. Opt.* **46**, 1819–29.
- 62 Sowa, Y., Rowe, A. D., Leake, M. C., Yakushi, T., Homma, M., Ishijima, A. and Berry, R. M. (2005) Direct observation of steps in rotation of the bacterial flagellar motor. *Nature* **437**, 916–9.
- 63 Reid, S. W., Leake, M. C., Chandler, J. H., Lo, C.-J., Armitage, J. P. and Berry, R. M. (2006) The maximum number of torque-generating units in the flagellar motor of *Escherichia coli* is at least 11. *Proc. Natl. Acad. Sci. U. S. A.* **103**, 8066–71.
- 64 Little, M. A., Steel, B. C., Bai, F., Sowa, Y., Bilyard, T., Mueller, D. M., Berry, R. M. and

- Jones, N. S. (2011) Steps and bumps: precision extraction of discrete states of molecular machines. *Biophys. J.* **101**, 477–485.
- 65 Yildiz, A. and Selvin, P. R. (2005) Fluorescence imaging with one nanometer accuracy: application to molecular motors. *Acc. Chem. Res.* **38**, 574–82.
- 66 Yildiz, A., Forkey, J. N., McKinney, S. A., Ha, T., Goldman, Y. E. and Selvin, P. R. (2003) Myosin V walks hand-over-hand: single fluorophore imaging with 1.5-nm localization. *Science* **300**, 2061–5.
- 67 Aubin-Tam, M.-E., Olivares, A. O., Sauer, R. T., Baker, T. A. and Lang, M. J. (2011) Single-molecule protein unfolding and translocation by an ATP-fueled proteolytic machine. *Cell* **145**, 257–67.
- 68 Shu, D., Zhang, H., Jin, J. and Guo, P. (2007) Counting of six pRNAs of phi29 DNA-packaging motor with customized single-molecule dual-view system. *EMBO J.* **26**, 527–537.
- 69 Leake, M. C., Chandler, J. H., Wadhams, G. H., Bai, F., Berry, R. M. and Armitage, J. P. (2006) Stoichiometry and turnover in single, functioning membrane protein complexes. *Nature* **443**, 355–8.
- 70 Svoboda, K., Schmidt, C. F., Schnapp, B. J. and Block, S. M. (1993) Direct observation of kinesin stepping by optical trapping interferometry. *Nature* **365**, 721–727.
- 71 Kerssemakers, J. W. J., Munteanu, E. L., Laan, L., Noetzel, T. L., Janson, M. E. and Dogterom, M. (2006) Assembly dynamics of microtubules at molecular resolution. *Nature*, Nature Publishing Group **442**, 709–12.
- 72 McGuire, H., Aourousseau, M. R. P., Bowie, D. and Bluncks, R. (2012) Automating single subunit counting of membrane proteins in mammalian cells. *J. Biol. Chem.* **287**, 35912–35921.
- 73 Das, S. K., Darshi, M., Cheley, S., Wallace, M. I. and Bayley, H. (2007) Membrane protein stoichiometry determined from the step-wise photobleaching of dye-labelled subunits. *ChemBioChem* **8**, 994–9.
- 74 Haran, G. (2004) Noise reduction in single-molecule fluorescence trajectories of folding proteins. *Chem. Phys.* **307**, 137–145.
- 75 Smith, D. A. (1998) A Quantitative Method for the Detection of Edges in Noisy Time-Series. *Philos. Trans. R. Soc., B* **353**, 1969–1981.
- 76 Reuel, N. F., Bojo, P., Zhang, J., Boghossian, A. A., Ahn, J.-H., Kim, J.-H. and Strano, M. S. (2012) NoRSE: noise reduction and state evaluator for high-frequency single event traces. *Bioinformatics* **28**, 296–297.
- 77 Chung, S. H. and Kennedy, R. A. (1991) Forward-backward non-linear filtering technique for extracting small biological signals from noise. *J. Neurosci. Methods* **40**, 71–86.
- 78 Arant, R. J. and Ulbrich, M. H. (2014) Deciphering the subunit composition of multimeric proteins by counting photobleaching steps. *ChemPhysChem* **15**, 600–605.
- 79 Ulbrich, M. H. and Isacoff, E. Y. (2007) Subunit counting in membrane-bound proteins. *Nat. Methods* **4**, 319–321.
- 80 Coffman, V. C., Wu, P., Parthun, M. R. and Wu, J.-Q. (2011) CENP-A exceeds microtubule attachment sites in centromere clusters of both budding and fission yeast. *J. Cell Biol.* **195**, 563–72.

- 81 Lee, I.-J. and Wu, J.-Q. (2012) Characterization of Mid1 domains for targeting and scaffolding in fission yeast cytokinesis. *J. Cell Sci.* **125**, 2973–85.
- 82 Lawrimore, J., Bloom, K. S. and Salmon, E. D. (2011) Point centromeres contain more than a single centromere-specific Cse4 (CENP-A) nucleosome. *J. Cell Biol.* **195**, 573–82.
- 83 Hines, K. E. (2013) Inferring subunit stoichiometry from single molecule photobleaching. *J. Gen. Physiol.* **141**, 737–46.
- 84 Coffman, V. C. and Wu, J.-Q. (2012) Counting protein molecules using quantitative fluorescence microscopy. *Trends Biochem. Sci.* **37**, 499–506.
- 85 Waters, J. C. (2009) Accuracy and precision in quantitative fluorescence microscopy. *J. Cell Biol.* **185**, 1135–48.
- 86 Shaner, N. C., Patterson, G. H. and Davidson, M. W. (2007) Advances in fluorescent protein technology. *J. Cell Sci.* **120**, 4247–60.
- 87 Tsien, R. Y. (1998) The green fluorescent protein. *Annu. Rev. Biochem., Annual Reviews* 4139 El Camino Way, P.O. Box 10139, Palo Alto, CA 94303-0139, USA **67**, 509–44.
- 88 Pédelacq, J.-D., Cabantous, S., Tran, T., Terwilliger, T. C. and Waldo, G. S. (2006) Engineering and characterization of a superfolder green fluorescent protein. *Nat. Biotechnol., Nature Publishing Group* **24**, 79–88.
- 89 Waldo, G. S., Standish, B. M., Berendzen, J. and Terwilliger, T. C. (1999) Rapid protein-folding assay using green fluorescent protein. *Nat. Biotechnol., Nature Publishing Group* **17**, 691–695.
- 90 Heim, R., Cubitt, A. B. and Tsien, R. Y. (1995) Improved green fluorescence. *Nature* **373**, 663–664.
- 91 Grubmayer, K. S., Kurz, A. and Herten, D.-P. (2014) Single-molecule studies on the label number distribution of fluorescent markers. *Chemphyschem* **15**, 734–42.
- 92 Amrute-Nayak, M., Lambeck, K.-A., Radocaj, A., Huhnt, H. E., Scholz, T., Hahn, N., Tsiavaliaris, G., Walter, W. J. and Brenner, B. (2014) ATP turnover by individual myosin molecules hints at two conformers of the myosin active site. *Proc. Natl. Acad. Sci.* **111**, 2536–2541.
- 93 Xie, X. S. and Lu, H. P. (1999) Single-molecule Enzymology. *J. Biol. Chem.* **274**, 15967–15970.
- 94 Oiwa, K., Eccleston, J. F., Anson, M., Kikumoto, M., Davis, C. T., Reid, G. P., Ferenczi, M. A., Corrie, J. E., Yamada, A., Nakayama, H., et al. (2000) Comparative single-molecule and ensemble myosin enzymology: sulfoindocyanine ATP and ADP derivatives. *Biophys. J.* **78**, 3048–71.
- 95 Bagshaw, C. R. and Conibear, P. B. (2000) Single-Molecule Enzymology: Critical Aspects Exemplified by Myosin ATPase Activity. *Single Mol.* **1**, 271–277.
- 96 Colquhoun, D. and Sigworth, F. J. (1995) Fitting and Statistical Analysis of Single-Channel Records. In *Single-Channel Recording* (Sakmann, B., and Neher, E., eds.) 2nd ed., pp 483–588, Plenum Press, New York.
- 97 Green, M. R. and Sambrook, J. (2012) *Molecular Cloning: A Laboratory Manual* (3 vols.) 4th ed., Cold Spring Harbor Laboratory Press, New York.
- 98 Studier, W. F. (2005) Protein production by auto-induction in high-density shaking cultures. *Protein Expr. Purif.* **41**, 207–234.

- 99 Selvin, P. R., Loughheed, T., Tonks Hoffman, M., Park, H., Balci, H., Blehm, B. H. and Toprak, E. (2008) In Vitro and In Vivo FIONA and Other Acronyms for Watching Molecular Motors Walk. In *Single-Molecule Techniques: A Laboratory Manual* (Selvin, P. R., and Ha, T., eds.), pp 37–72, Cold Spring Harbor Laboratory Press, New York.
- 100 Kielkopf, C. L., Bauer, W. and Urbatsch, I. L. (2012) Expressing Cloned Genes for Protein Production, Purification, and Analysis. In *Molecular Cloning: A Laboratory Manual* (3 vols.) 4th ed., pp 1481–1636, Cold Spring Harbor Laboratory Press, New York.
- 101 Ladner, C. L., Yang, J., Turner, R. J. and Edwards, R. A. (2004) Visible fluorescent detection of proteins in polyacrylamide gels without staining. *Anal. Biochem.* **326**, 13–20.
- 102 Magrane, M. and UniProt Consortium. (2011) UniProt Knowledgebase: a hub of integrated protein data. *Database* **2011**, bar009.
- 103 Johnson, I. (Ed.). (2010) *The Molecular Probes® Handbook: A Guide to Fluorescent Probes and Labeling Technologies* 11th ed., Life Technologies Corporation.
- 104 Engel, P. C. (1996) Enzyme Cofactors. In *Enzymology LabFax* (Engel, P. C., ed.), pp 223–248, BIOS Scientific Publishers Limited, Oxford.
- 105 Falk, J. E. (1964) Porphyrins and Metalloporphyrins: Their General, Physical and Coordination Chemistry, and Laboratory Methods. *Biochem. Biophys. Acta Libr.* Vol. 2, Elsevier, London.
- 106 Bradford, M. M. (1976) A rapid and sensitive method for the quantitation of microgram quantities of protein utilizing the principle of protein-dye binding. *Anal. Biochem.* **72**, 248–254.
- 107 Kuzmic, P. (1996) Program DYNAFIT for the Analysis of Enzyme Kinetic Data: Application to HIV Proteinase. *Anal. Biochem.* **237**, 260–273.
- 108 Tachibana, S., Suzuki, M. and Asano, Y. (2006) Application of an enzyme chip to the microquantification of L-phenylalanine. *Anal. Biochem.* **359**, 72–78.
- 109 Kern, W. and Puotinen, D. A. (1970) Cleaning solutions based on hydrogen peroxide for use in silicon semiconductor technology. *RCA Rev.* **31**, 187–206.
- 110 Inverted Microscope Eclipse Ti-E Ti-E/B Instructions, Nikon, Kingston Upon Thames.
- 111 Peters, J. (2008) Nikon Instruments TiE-PFS Dynamic Focusing System. *Nat. Methods | Appl. Notes*, Nature Publishing Group.
- 112 Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., et al. (2012) Fiji: an open-source platform for biological-image analysis. *Nat. Methods* **9**, 676–682.
- 113 Turner, R. D., Hurd, A. F., Cadby, A., Hobbs, J. K. and Foster, S. J. (2013) Cell wall elongation mode in Gram-negative bacteria is determined by peptidoglycan architecture. *Nat. Commun.* **4**, 1496.
- 114 Betzig, E., Patterson, G. H., Sougrat, R., Lindwasser, O. W., Olenych, S., Bonifacino, J. S., Davidson, M. W., Lippincott-Schwartz, J. and Hess, H. F. (2006) Imaging intracellular fluorescent proteins at nanometer resolution. *Science* **313**, 1642–1645.
- 115 Hermanson, G. (2008) *Bioconjugate Techniques* 2nd Editio., Academic Press, London.
- 116 Mashanov, G. I., Nobles, M., Harmer, S. C., Molloy, J. E. and Tinker, A. (2010) Direct

- observation of individual KCNQ1 potassium channels reveals their distinctive diffusive behavior. *J. Biol. Chem.* **285**, 3664–3675.
- 117 Madl, J., Weghuber, J., Fritsch, R., Derler, I., Fahrner, M., Frischauf, I., Lackner, B., Romanin, C. and Schütz, G. J. (2010) Resting state Orai1 diffuses as homotetramer in the plasma membrane of live mammalian cells. *J. Biol. Chem.* **285**, 41135–41142.
 - 118 Schmidt, T., Schutz, G. J., Gruber, H. J. and Schindler, H. (1996) Local stoichiometries determined by counting individual molecules. *Anal. Chem.* **68**, 4397–4401.
 - 119 Gordon, M. P., Ha, T. and Selvin, P. R. (2004) Single-molecule high-resolution imaging with photobleaching. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 6462–5.
 - 120 Pardo, A. D., Chereskin, B. M., Castelfranco, P. A., Franceschi, V. R. and Wezelman, B. E. (1980) ATP Requirement for Mg Chelatase in Developing Chloroplasts. *PLANT Physiol.* **65**, 956–960.
 - 121 Ye, J. Y., Yamauchi, M., Yogi, O. and Ishikawa, M. (1999) Spectroscopic Properties of 2'-(or-3') -O- (2,4,6-Trinitrophenyl) Adenosine 5'-Triphosphate Revealed by Time-Resolved Fluorescence Spectroscopy. *J. Phys. Chem. B, American Chemical Society* **103**, 2812–2817.
 - 122 Hiratsuka, T. (1982) Biological activities and spectroscopic properties of chromophoric and fluorescent analogs of adenine nucleoside and nucleotides, 2',3'-O-(2,4,6-trinitrocyclohexadienylidene) adenosine derivatives. *Biochim. Biophys. Acta - Gen. Subj.* **719**, 509–517.
 - 123 Grubmeyer, C. and Penefsky, H. S. (1981) The presence of two hydrolytic sites on beef heart mitochondrial adenosine triphosphatase. *J. Biol. Chem.* **256**, 3718–27.
 - 124 Rai, S. S. and Kasturi, S. R. (1994) Interaction of TNP-ATP with tubulin: A fluorescence spectroscopic study. *Biophys. Chem.* **48**, 359–368.
 - 125 Bendorowicz-Pikuła, J. A nucleotide-binding domain of porcine liver annexin VI. Proteolysis of annexin VI labelled with 8-azido-ATP, purification by affinity chromatography on ATP-agarose, and fluorescence studies. *Mol. Cell. Biochem.*, Kluwer Academic Publishers **181**, 11–20.
 - 126 Bagshaw, C. (2001) ATP analogues at a glance. *J. Cell Sci.* **114**, 459–460.
 - 127 Webb, M. R. and Corrie, J. E. (2001) Fluorescent coumarin-labeled nucleotides to measure ADP release from actomyosin. *Biophys. J., Elsevier* **81**, 1562–9.
 - 128 Kuhlman, P. a. and Bagshaw, C. R. (1998) ATPase kinetics of the Dictyostelium discoideum myosin II motor domain. *J. Muscle Res. Cell Motil.* **19**, 491–504.
 - 129 Zhao, C., Matveeva, E. A., Ren, Q. and Whiteheart, S. W. (2010) Dissecting the N-ethylmaleimide-sensitive factor: required elements of the N and D1 domains. *J. Biol. Chem.* **285**, 761–72.
 - 130 Schlee, S., Groemping, Y., Herde, P., Seidel, R. and Reinstein, J. (2001) The chaperone function of ClpB from *Thermus thermophilus* depends on allosteric interactions of its two ATP-binding sites. *J. Mol. Biol.* **306**, 889–99.
 - 131 Hattendorf, D. A. and Lindquist, S. L. (2002) Cooperative kinetics of both Hsp104 ATPase domains and interdomain communication revealed by AAA sensor-1 mutants. *EMBO J.* **21**, 12–21.
 - 132 Vogt, A. D. and Di Cera, E. (2012) Conformational Selection or Induced Fit? A Critical Appraisal of the Kinetic Mechanism. *Biochemistry* **51**, 5894–5902.
 - 133 Castelfranco, P. A., Weinstein, J. D., Schwarcz, S., Pardo, A. D. and Wezelman, B. E.

- (1979) The Mg insertion step in chlorophyll biosynthesis. *Arch. Biochem. Biophys.* **192**, 598–592.
- 134 Stöckel, J. and Oelmüller, R. (2004) A novel protein for photosystem I biogenesis. *J. Biol. Chem.* **279**, 10243–10251.
- 135 Waterhouse, A. M., Procter, J. B., Martin, D. M. A., Clamp, M. and Barton, G. J. (2009) Jalview Version 2-a multiple sequence alignment editor and analysis workbench. *Bioinformatics* **25**, 1189–91.
- 136 Larkin, M. A., Blackshields, G., Brown, N. P., Chenna, R., McGettigan, P. A., McWilliam, H., Valentin, F., Wallace, I. M., Wilm, A., Lopez, R., et al. (2007) Clustal W and Clustal X version 2.0. *Bioinformatics* **23**, 2947–8.

Appendix A: MATLAB code for localization of fluorophores

These functions were written by Robert Turner[113] (MBB, The University of Sheffield) using photoactivation localization microscopy/STORM methodology as described previously by Betzig et al.[114] They were minimally adapted to purpose in the present work. A list of the arguments is provided with explanations for each user-manipulated function. See Section 4.1. for an explanation of their implementation. Functions called by functions but not requiring user input are given at the end without explanatory notes.

LM_filelist

Inputs

folder: the file path specifying the folder holding the individual TIF format images of the image series

Outputs

filelist: a cell array holding the file path to each image file in the series

Custom functions called by LM_filelist

LM_osslash

```
function [ filelist ] = LM_filelist( folder )
%LM_FILELIST Summary of this function goes here
% Detailed explanation goes here

%slash direction
slash=LM_osslash;

files=dir(folder);

nfiles=size(files,1);

filelist=cell(nfiles,1);

%filelist=cellstr(filelist);

for i=1:nfiles;
    if (not(isempty(strfind(files(i).name,'tif'))))
        filelist{i}=strcat(folder, slash, files(i).name);
    end
end

filelist=filelist(cellfun('length',filelist)>0);

end
```

LM_MakeReconset

Fields

threshold: the greyscale intensity value above which a pixel will register as a putative fluorophore

spacing: the minimum number of pixels which must separate the centres of two putative fluorophores in order that they register as such

log_width: the standard deviation of the Laplacian of Gaussian filter used to filter the images

erode: specifies the radius of the disk structuring element used to erode the thresholded image to reduce noise

dilate: specifies the radius of the disk structuring element used to dilate the eroded image to restore the thresholded image after noise reduction

graphics: indicates whether to display the graphics associated with a particular function (set to 0 to suppress graphics or 1 to display graphics)

filter: indicates whether to pass the image through a Laplacian of Gaussian filter in the function LM_reconstruct (set to 0 to skip the filter or 1 to apply the filter)

displaymin: specifies the minimum intensity value to be used as the lower bound of the intensity range when displaying a raw image

displaymax: specifies the maximum intensity value to be used as the upper bound of the intensity range when displaying a raw image

averageframes: number of images over which to average in the current series if a reconstruction of block averages is desired

subtractmin: indicates whether to create a 'minimum image' (takes a sample of 100 images across the image series and returns the minimum intensity value found for each pixel) and subtract it from each image in the series as a form of background removal (set to 0 to skip minimum image subtraction or 1 to apply it)

Outputs

reconset: struct containing the fields and values specified in LM_MakeReconset

```
function [ reconset ] = LM_MakeReconset( )
%UNTITLED Summary of this function goes here
% Detailed explanation goes here

reconset.threshold=90000;
reconset.spacing=5;
reconset.log_width=1;
reconset.erode=2;
reconset.dilate=1;
reconset.graphics=0; %0=off, 1=on
reconset.filter=0; %0=off, 1=on
reconset.displaymin=2000;
reconset.displaymax=5000;
reconset.averageframes=1;
reconset.subtractmin=0; %0=off, 1=on

end
```

LM_reconstruct

Inputs

filelist: cell array holding the file path to each image file in the series, created by the function LM_filelist

reconset: struct containing the fields and values specified in the function LM_MakeReconset

images: optional argument, m by n by p array containing the image series of m images with dimensions n by p pixels

Outputs

candidates: matrix containing information on the fluorophores identified and localized by LM_reconstruct, each row being a separate entry (see below for the column key)

Key to columns in candidates matrix

- 1: intensity
- 2: x position
- 3: x width
- 4: y position
- 5: y width
- 6: local background
- 7: frame number
- 8: piezo position at capture
- 9: z position
- 10: emitter identifier
- 11: deepfield step number
- 12: deepfield z position
- 13: normalized residual of fit
- 14: curve fit exit flag
- 15: candidate (bright pixel) x position
- 16: candidate (bright pixel) y position
- 17: nearest neighbour (2D) in nm
- 18: nearest neighbour (3D) in nm
- 19: x position relative to cutout
- 20: y position relative to cutout
- 21: Channel

Custom functions called by LM_reconstruct

LM_osslash

LM_avimages

LM_fit

```
function [ candidates ] = LM_reconstruct( filelist, reconset,
images )
%UNTITLED Summary of this function goes here
% Detailed explanation goes here

runtime=0;

figure_wait=waitbar(0,'Initialising...','name','LM
Reconstruction','CreateCancelBtn',...
'setappdata(gcf,'canceling',1)');
movegui(figure_wait,'northwest');

if reconset.graphics==1
    fits_overlay=figure;
end
```

```

tempfits=[];
candidates=[];
extracts=[];

nfiles=size(filelist,1);

%if data is not in ram to start with, a sample of images must
be loaded
%to obtain dimensions and a minimum image to subtract when
filtering
if (nargin==2 && reconset.subtractmin==1)
    j=0;
    for i=1:round(nfiles/100):nfiles
        j=j+1;
        filepath=filelist{i};
        images(j, :, :) = double(imread(char(filepath)));
    end
    minimage=squeeze(min(images));
else
    %open up an image just to check dimensions
    images(1, :, :) = double(imread(char(filelist{1})));
end

dimension_y=size(images,2);
dimension_x=size(images,3);

%used later for graphics
myxlim=[0 dimension_x];
myylim=[0 dimension_y];

%slash direction
slash=LM_oslash;

%ensure, if using block averages, that the routine stops
before we run
%out of files.

nfiles=floor(nfiles/reconset.averageframes)*reconset.averageframes
;

for i=1:reconset.averageframes:nfiles
    tic;
    %get metadata
    filepath=filelist{i};
    [folder,filenameonly,~]=fileparts(filepath);
    metafile=strcat(folder, slash, filenameonly, '_Z.tsv');
    if exist(metafile, 'file')
        meta=double(dlmread(char(metafile)));
        pz=meta(1);
        if size(meta,2)>1
            step=meta(4);
        end
    else
        pz=-1; %set to zero if there is no meta file
        step=-1; %set to zero if there is no meta file
    end

    %get framenumbers
    framenumbers=str2double(regexpi(filenameonly, '[0-

```



```

9]+'match'));

%get data
filepath=filelist{i};

if nargin==3
    %files in memory
    rawimage=squeeze(images(i,:,:));
elseif reconset.averageframes>1;
    %reconstruction of block averages
rawimage=LM_avimages(filelist(i:i+reconset.averageframes-1));
else
    %normal reconstruction
    rawimage=double(imread(char(filepath)));
end

if reconset.filter==1

    if reconset.subtractmin==1
        %subtract minimum values
        filteredimage=rawimage-minimage;
    else
        filteredimage=rawimage;
    end

    %apply Laplacian of Gaussian filter
    LOG = fspecial('log',[10 10],reconset.log_width);
    filteredimage = -
1*imfilter(filteredimage,LOG,'replicate');
    activeimage=filteredimage;
else
    activeimage=rawimage;
end

%threshold and erode
binimage=activeimage>reconset.threshold;
%{
se = strel('disk',reconset.erode);
erodedimage=imerode(binimage,se);
se = strel('disk',reconset.dilate);
erodedimage=imdilate(erodedimage,se);
%}
erodedimage=binimage;

%identify "grains"
%grainimage = bwlabel(erodedimage);

%identify centres of mass (and fit)
brightpixels =
regionprops(erodedimage,rawimage,'WeightedCentroid');
%can change to parfor
parfor j=1:size(brightpixels,1)
    %fitting block co-ordinates
    min_coords=round(brightpixels(j).WeightedCentroid)-
reconset.spacing;

    max_coords=round(brightpixels(j).WeightedCentroid)+reconset.spacin

```

```

g;

myfit=[-1 -1 -1 -1 -1 -1 -1 -1];

rawx=-1;
rawy=-1;

%check if too close to edge of frame
if (min(min_coords)>0 && max_coords(1,1)<dimension_x
&& max_coords(1,2)<dimension_y)

extract=rawimage(min_coords(2):max_coords(2),min_coords(1):max_coords(1));

myfit=LM_fit(extract);

%log positions relative to cutout reference frame
%(experimental)
rawx=myfit(2);
rawy=myfit(4);

%translate xy position to main reference frame
(fitting method)
%the -1 is there because in the fit pixel index 1
is
%treated as pixel index 0
myfit(2)=myfit(2)+min_coords(1)-1;
myfit(4)=myfit(4)+min_coords(2)-1;

%per fit diagnostics
%{
hold on

    imagesc(rawimage);
    set(gca,'DataAspectRatio',[1 1 1]);
    set(gca,'YDir','reverse')
    scatter(myfit(:,2),myfit(:,4),'k');

hold off
%}

end

afit=[myfit(1:6) framenumbers pz*1000 0 0 step 0
myfit(7:8) brightpixels(j).WeightedCentroid];
afit(19:20)=[rawx rawy];

tempfits=[tempfits; afit];

%tempfits(j,1:16)=[myfit(1:6) framenumbers pz*1000 0 0
step 0 myfit(7:8) brightpixels(j).WeightedCentroid];

%tempfits(j,19)=rawx;
%tempfits(j,20)=rawy;

end

```

```

    if size(tempfits,1)>0
        candidates=[candidates; tempfits];
    end

    %blinkimage=activeimage.*erodedimage;

    if reconset.graphics==1
        clf;
        hold on;
        figure(fits_overlay)
        %imagesc(rawimage,[reconset.displaymin
reconset.displaymax]);
        imagesc(filteredimage);
        set(gca,'DataAspectRatio',[1 1 1]);
        set(gca,'YDir','reverse')
        title(strcat('Raw File:
',int2str(i),'/',int2str(nfiles)),'FontWeight','bold');
        colormap('gray');

        B=bwboundaries(erodedimage);
        for k = 1:length(B)
            boundary = B{k};
            patch(boundary(:,2)+0.5,
boundary(:,1)+0.5,'g','FaceAlpha',0.3);
        end
        if size(tempfits,1)>0
            scatter(tempfits(:,2),tempfits(:,4));
        end
        hold off;
        xlim(myxlim);
        ylim(myylim);
        drawnow;
        myxlim=xlim;
        myylim=ylim;
    end

    %{
    if reconset.graphics==1
        subplot(2,3,1)
        imagesc(rawimage,[reconset.displaymin
reconset.displaymax]);
        set(gca,'DataAspectRatio',[1 1 1]);
        title(strcat('Raw File:
',int2str(i),'/',int2str(nfiles)),'FontWeight','bold');

        subplot(2,3,2)
        imagesc(activeimage, [mean(mean(activeimage))
reconset.threshold*10]);
        set(gca,'DataAspectRatio',[1 1 1]);
        title(strcat('Filtered'),'FontWeight','bold');

        subplot(2,3,3)
        imagesc(grainimage);
        set(gca,'DataAspectRatio',[1 1 1]);
        title(strcat('Events'),'FontWeight','bold');

        subplot(2,3,4)

```

```

        if size(candidates,1)>0
            scatter(candidates(:,15),candidates(:,16),2);
            set(gca,'DataAspectRatio',[1 1 1]);
            axis([0 dimension_x 0 dimension_y])
            set(gca,'YDir','reverse')
            title(strcat('Centre of
Mass'),'FontWeight','bold');
        end

        subplot(2,3,5)
        if size(candidates,1)>0
            scatter(candidates(:,2),candidates(:,4),2);
            set(gca,'DataAspectRatio',[1 1 1]);
            axis([0 dimension_x 0 dimension_y])
            set(gca,'YDir','reverse')
            title(strcat('Fit'),'FontWeight','bold');
        end

        drawnow;
    end
    %}

    %empty temporary variables
    tempfits=[];

    runtime=runtime+toc;
    meanframetime=runtime/i;
    remaintime=((nfiles-i)*meanframetime)/60;
    waitbar(i/nfiles,figure_wait,strcat('Mean time per frame
(s): ',num2str(meanframetime,5),' Time left (mins): ',
num2str(remaintime,5)));

    % Check for Cancel button press
    if getappdata(figure_wait,'canceling')
        break
    end

end

%remove rows marked with "-1" i.e. no fit returned
candidates=candidates(candidates(:,1)>-1,:);

%get rid of visualisations
delete(figure_wait); %waitbar
%{
if reconset.graphics==1
    delete(fits_overlay);
end
%}

end

```

LM_emitters

Inputs

events: matrix containing information on the fluorophores identified and localized by LM_reconstruct, each row being a separate entry (see LM_reconstruct for the column key)

dx: maximum spacing in the x-direction allowed for two localizations to be considered as the same fluorophore

dy: maximum spacing in the y-direction allowed for two localizations to be considered as the same fluorophore

dz: maximum spacing in the z-direction allowed for two localizations to be considered as the same fluorophore

dframe: maximum number of frames allowed between two localizations for them to be considered as the same fluorophore

Outputs

emitters: matrix containing information on the fluorophore localizations, including newly assigned fluorophore identifier numbers (see LM_reconstruct for the column key)

```
function [ emitters ] = LM_emitters( events, dx, dy, dz, dframe )
%LM_NEIGHBOURS Summary of this function goes here
% Detailed explanation goes here

runtime=0;

figure_wait=waitbar(0,'Initialising...','name','LM
Reconstruction','CreateCancelBtn',...
    'setappdata(gcf,'canceling',1)');
movegui(figure_wait,'northwest');

emitters=[];

events(:,10)=0;

j=1;

n=size(events,1);

for i=1:n
    tic;
    %x<x1+dx
    %x>x1-dx
    %y<y1+dy
    %y>y1-dy
    %frame<frame1+dframe
    %frame>frame1
    %emitter not set
    if(events(i,10)==0)
        events(i,10)=j;
        j=j+1;
    end

    linkedrows= events(:,2)<events(i,2)+dx &
events(:,2)>events(i,2)-dx & ...
```

```

        events(:,4)<events(i,4)+dy & events(:,4)>events(i,4)-dy &
...
        events(:,9)<events(i,9)+dz & events(:,9)>events(i,9)-dz &
...
        events(:,7)>events(i,7) & events(:,7)<=events(i,7)+dframe
& ...
        events(:,10)<1;

        events(linkedrows,10)=events(i,10);

        runtime=runtime+toc;
        meanframetime=runtime/i;
        remaintime=((n-i)*meanframetime)/60;
        waitbar(i/n,figure_wait,strcat('Mean time per localisation
(s): ',num2str(meanframetime,5),' Time left (mins):',
num2str(remaintime,5)));

        % Check for Cancel button press
        if getappdata(figure_wait,'canceling')
            break
        end
    end

    emitters=events;

    %get rid of visualisations
    delete(figure_wait); %waitbar

end

```

LM_osslash

```

function [ slash ] = LM_osslash()
%LM_OSSLASH Summary of this function goes here
%   Detailed explanation goes here

    %gets path delimiting slash character for current operating
system
    if (strcmp(getenv('os'),'Windows_NT')==1)
        slash='\';
    else
        slash='/';
    end

end

```

LM_avimages

```
function [ avimage ] = LM_avimages( filelist )
%UNTITLED3 Summary of this function goes here
% Detailed explanation goes here

nfiles=size(filelist,1);

for i=1:nfiles
    filepath=filelist{i};
    images(i,:,:)=double(imread(char(filepath)));
end

avimage=squeeze(mean(images));

end
```

LM_fit

Custom functions called by LM_fit

Gaussian2DJ

```
function [ params ] = LM_fit( extract )
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here

ex_dim=size(extract,1);

%scale extract so intensities are roughly on a range of 0 to
the dimensions of the region
%this makes the problem "well scaled": all variables vary
within
%similar ranges

raw_ex_max=max(max(extract));

extract=(extract./raw_ex_max).*ex_dim;

ex_max=max(max(extract));
ex_min=min(min(extract));
ex_av=mean(mean(extract));

%set up variables for fitting
XY=ex_dim;
[X,Y]=meshgrid(1:XY,1:XY); %your x-y coordinates
x(:,1)=X(:); % x= first column
x(:,2)=Y(:); % y= second column

lower=[0 0 0 0 0 0];
guesses=[ex_max ex_dim/2 ex_dim/2 ex_dim/2 ex_dim/2 ex_min];
upper=[ex_max*2 ex_dim ex_dim ex_dim ex_dim ex_min*2];

%Check upper bounds
neg=find(upper==0);
if ~isempty(neg)
```

```

        upper(neg)=1e-6;
    end

    %options=optimset('Display','off');
    options=optimset('Jacobian','on','Display','off','TolFun',1e-
6);

[bestfit,resn,~,ef]=lsqcurvefit(@Gaussian2DJ,guesses,x,extract(:),
lower,upper,options);

    %scale intensity and background values back to normal
    bestfit(1)=(bestfit(1)/ex_dim)*raw_ex_max;
    bestfit(6)=(bestfit(6)/ex_dim)*raw_ex_max;

    params=[bestfit resn ef];

    %extractspline=interp2(extract,3,'cubic');

    %per fit diagnostics
    %{
    hold on

        imagesc(extract);
        set(gca,'DataAspectRatio',[1 1 1]);
        set(gca,'YDir','reverse')
        scatter(bestfit(:,2),bestfit(:,4));

    hold off
    %}
end

```

Gaussian2DJ

```

function [F, J] = Gaussian2DJ(p,xdata)

a=p(1);
b=p(2);
c=p(3);
d=p(4);
e=p(5);
f=p(6);

x=xdata(:,1);
y=xdata(:,2);

F = (a.*exp(-1.*(((x-b)/c).^2)+((y-d)/e).^2))+f;

Ja=exp(-(b-x).^2/c.^2-(d-y).^2/e.^2);
Jb=-(a.*exp(-(b-x).^2/c.^2-(d-y).^2/e.^2)).*(2.*b-
2.*x)/c.^2;
Jc=(2.*a.*exp(-(b-x).^2/c.^2-(d-y).^2/e.^2)).*(b-
x).^2/c.^3;
Jd=-(a.*exp(-(b-x).^2/c.^2-(d-y).^2/e.^2)).*(2.*d-
2.*y)/e.^2;
Je=(2.*a.*exp(-(b-x).^2/c.^2-(d-y).^2/e.^2)).*(d-

```



```
y).^2)/e.^3;  
Jf(1:size(xdata,1),1)=1;  
  
J = [Ja, Jb, Jc, Jd, Je, Jf];  
  
end
```


Appendix B: MATLAB code for analysis of stepwise photobleaching intensity traces

A list of the arguments is provided with explanations for each user-manipulated function. See Section 4.2. for an explanation of their implementation. Functions called by functions but not requiring user input are given at the end without explanatory notes.

extract_avg_xy

Inputs

events: matrix containing information on the fluorophore localizations (see LM_reconstruct in Appendix A: for the column key)

Outputs

avg_xy: matrix containing fluorophore identifier numbers and average positions in column format (column 1: identifier number; column 2: average x-coordinate; column 3: average y-coordinate)

```
function [ avg_xy ] = extract_avg_xy( events )
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
% Extract xy values and emitter id number from events table.
% col #1 - emitter id number
% col #2 - avg x position
% col #3 - avg y position

if events(1,10)==0
    no_em=size(events,1);
    avg_xy=zeros(no_em,3);
    avg_xy(:,1)=1:no_em;
    avg_xy(:,2)=events(:,2);
    avg_xy(:,3)=events(:,4);
else
    no_em=max(events(:,10));
    avg_xy=zeros(no_em,3);
    avg_xy(:,1)=1:no_em;

    for id=1:no_em
        rows=events(:,10)==id;
        avg_xy(id,2)=mean(events(rows,2));
        avg_xy(id,3)=mean(events(rows,4));
    end
end

end
```

check_sep

Inputs

reconset: struct containing the fields and values specified in LM_MakeReconset (see Appendix A:)

events: matrix containing information on the fluorophore localizations (see LM_reconstruct in Appendix A: for the column key)

filelist: cell array holding the file path to each image file in the series, created by the function LM_filelist (see Appendix A:)

avg_xy: optional argument, only included if processing localizations from a single frame, or drift is negligible enough to allow use of average x- and y-positions (generated by the function `extract_avg_xy`)

Outputs

events_new: matrix containing information on the fluorophore localizations, excluding those fluorophores deemed as overlapping (see `LM_reconstruct` for the column key)

avg_xy_new: only returned if `avg_xy` is supplied as input; matrix with the same structure as `avg_xy`, but excluding those fluorophores deemed as overlapping

```
function [ events_new, avg_xy_new ] = check_sep( reconset, events,
filelist, avg_xy )
%CHECK_SEP Removes fluorophores occurring within the separation
distance of
%each other. Assumes local background subtraction method, so adds
1 pixel
%to the separation value. Only include avg_xy if processing fits
from a
%single frame. If processing fits from multiple frames, it only
removes
%fluorophores spatially overlapping simultaneously (i.e. in the
same
%frame).
% Choose to plot retained and discarded localisations by setting
% reconset.graphics=1.
% Detailed explanation goes here

separation=reconset.spacing+1;

if nargin==4

x=avg_xy(:,2);
y=avg_xy(:,3);
fluor_num=avg_xy(:,1);
close=0;

for data=1:length(x)

    ind_samex1=find(x>(x(data)-separation));
    ind_samex2=find(x<(x(data)+separation));
    ind_samex=intersect(ind_samex1,ind_samex2);
    omit= ind_samex==data;
    ind_samex(omit)=[];

    if ~isempty(ind_samex)
        for dum=ind_samex'
            if (y(dum)>(y(data)-separation) &&...
                y(dum)<(y(data)+separation))
                close=[close;dum];
            end
        end
    end
end

end

%remove starting zero & repeated index values
```

```

close(1)=[];
close=unique(close);

fluor_num(close)=[];
x(close)=[];
y(close)=[];

    if events(1,10)==0
        events(:,10)=1:size(events,1);
    end
    events(close,:)=[];
    events_new=events;

avg_xy_new(:,1)=fluor_num;
avg_xy_new(:,2)=x;
avg_xy_new(:,3)=y;

else
    no_frames=max(events(:,7))-min(events(:,7))+1;
    start_frame=min(events(:,7));
    close_fluors=0;
    for frame=1:no_frames
        rows=events(:,7)==start_frame+frame-1;
        extract=events(rows,:);
        close=0;
        for fluor=1:size(extract,1)
            ind_samex1=find(extract(:,2)>(extract(fluor,2)-
separation));
            ind_samex2=find(extract(:,2)<(extract(fluor,2)+separation));
            ind_samex=intersect(ind_samex1,ind_samex2);
            omit= ind_samex==fluor;
            ind_samex(omit)=[];

            if ~isempty(ind_samex)
                for dum=ind_samex'
                    if (extract(dum,4)>(extract(fluor,4)-separation) &&...
                        extract(dum,4)<(extract(fluor,4)+separation))
                        close=[close;dum];
                    end
                end
            end
        end

        %remove starting zero & repeated index values
close(1)=[];
close=unique(close);

close_fluors=[close_fluors;extract(close,10)];
    end

    close_fluors(1)=[];
    close_fluors=unique(close_fluors);

    if ~isempty(close_fluors)
        del_rows=0;
        xy_find=0;
        for dum=close_fluors'
            del=find(events(:,10)==dum);

```

```

        xy=find(events(:,10)==dum,1,'first');
        xy_find=[xy_find;xy];
        del_rows=[del_rows;del];
    end
    del_rows(1)=[];
    xy_find(1)=[];
    del_xy=[events(xy_find,2),events(xy_find,4)];
    events(del_rows,:)=[];
    else del_xy=0;
    end

    events_new=events;

    if reconset.graphics==1
        [~, keep_rows, ~]=unique(events_new(:,10),'first');
        keep_xy=[events_new(keep_rows,2),events_new(keep_rows,4)];
        [ignore, fits_overlay] = log_background_removal( filelist,
reconset, 1 );
        figure(fits_overlay)
        hold on;
        scatter(keep_xy(:,1),keep_xy(:,2),[],'b')
        if del_xy~=0
            scatter(del_xy(:,1),del_xy(:,2),[],'r')
        end
        hold off;
    end

end

end

```

choose_emitters

Inputs

events_new: matrix containing information on the fluorophore localizations, as returned by the function check_sep (see LM_reconstruct in Appendix A: for the column key)

nframes: scalar value; fluorophores localized within the first nframes of the image series are retained

filelist: cell array holding the file path to each image file in the series, created by the function LM_filelist (see Appendix A:)

reconset: struct containing the fields and values specified in LM_MakeReconset (see Appendix A:)

Outputs

events_new: matrix containing information on the fluorophore localizations, excluding fluorophores which appear for the first time after the first nframes of the image series (see LM_reconstruct for the column key)

Custom functions called by choose_emitters

log_background_removal

```

function [ events_new ] = choose_emitters( events_new, nframes,
filelist, reconset )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here

% Only keep emitters occurring in the first nframes specified

start_frame=events_new(1,7);
end_frame=start_frame+nframes-1;
events_new=sortrows(events_new,7);
end_row=find(events_new(:,7)==end_frame,1,'last');
keep_emitters=unique(events_new(1:end_row,10));
first_em=max(keep_emitters)+1;
last_em=max(events_new(:,10));
discard_emitters=first_em:last_em;

keep_rows=0;
for num=keep_emitters'
    rows=find(events_new(:,10)==num);
    keep_rows=[keep_rows;rows];
end

del_rows=0;
for num=discard_emitters
    rows=find(events_new(:,10)==num,1,'first');
    del_rows=[del_rows;rows];
end

del_rows(1)=[];
del_xy=[events_new(del_rows,2),events_new(del_rows,4)];

keep_rows(1)=[];
events_new=events_new(keep_rows,:);

[~, keep_rows, ~]=unique(events_new(:,10),'first');
    keep_xy=[events_new(keep_rows,2),events_new(keep_rows,4)];
    [ignore, fits_overlay] = log_background_removal( filelist,
reconset, 1 );
    figure(fits_overlay)
    hold on;
    scatter(keep_xy(:,1),keep_xy(:,2),[],'b')
    scatter(del_xy(:,1),del_xy(:,2),[],'r')
    hold off;

end

```

extract_int_local

Inputs

filelist: cell array holding the file path to each image file in the series (un-summed original image series in cases where localizations were performed on images created from several raw files), created by the function LM_filelist (see Appendix A:)

fits: matrix of fluorophore data, as returned by the function choose_emitters (see LM_reconstruct in Appendix A: for the column key)

reconset: struct containing the fields and values specified in LM_MakeReconset (see Appendix A: ; the reconset.averageframes field is taken to mean the number of frames of the original series which were summed or averaged to make one frame of the series used for localizations in the context of this function)

sub_back: indicates whether to perform local background removal on the intensity value for each fluorophore (set to 0 to skip background removal, or 1 to apply it)

Outputs

x_pos: m by n matrix containing x-coordinates for all n fluorophores in all m frames

y_pos: m by n matrix containing y-coordinates for all n fluorophores in all m frames

fluor_nos: column vector containing fluorophore identifier numbers

intensities: m by n matrix containing intensity values (background-corrected if chosen) for all n fluorophores in all m frames

fluor_det: m by n matrix of ones and zeros indicating whether a fluorophore was detected by the function LM_reconstruct for all n fluorophores in all m frames (1: fluorophore detected; 0: fluorophore not detected)

```
function [ x_pos, y_pos, fluor_nos, intensities, fluor_det ] =  
extract_int_local( filelist, fits, reconset, sub_back )  
%UNTITLED Summary of this function goes here  
% Detailed explanation goes here  
% Extract fluorophore intensities using individual  
% x,y positions for each  
% fluorophore in every frame. Employs optional local background  
subtraction (1 pixel  
% surrounding box) of the mean immediate background; set sub_back  
= 1 for  
% background subtraction, or 0 if not.  
  
%fluor_det = a series of 0 or 1  
%indicating if a frame was originally identified as having the  
fluorophore  
%on in it (1) or not (0) to allow to determine if an intensity  
increase is  
%due to the fluorophore of interest, or a neighbouring  
fluorophore. In  
%frames where the fluorophore doesn't appear, position is  
estimated from  
%previous/next position.  
%Also get x and y positions for each fluorophore in every frame  
%fluor_nos = number of each fluorophore  
  
%sum_step=no of frames original series was summed over  
  
separation=reconset.spacing;  
sum_step=reconset.averageframes;
```



```

fits=sortrows(fits,7);
fits_e_rat=fits;

nfiles=size(filelist,1);
fluor_nos=unique(fits_e_rat(:,10));
nfluors=length(fluor_nos);
x_pos=zeros(nfiles,nfluors);
y_pos=zeros(nfiles,nfluors);
intensities=zeros(nfiles,nfluors);
fluor_det=zeros(nfiles,nfluors);
boxsize=floor(separation/2);

%create x,y coordinates for fluorophores frame by frame
for fluor=1:nfluors

    rows=fits_e_rat(:,10)==fluor_nos(fluor);
    sample=fits_e_rat(rows,:);

    if fits(1,7)~=1
        sample(:,7)=sample(:,7)-fits(1,7)+1;
    end

    %project first position backwards to starting frame
    interval=sample(1,7)*sum_step;
    x_pos(1:interval,fluor)=sample(1,2);
    y_pos(1:interval,fluor)=sample(1,4);
    %project each position backwards by the summing interval
    if size(sample,1)>1
        for sumframe=1:size(sample,1)-1

            x_pos((sample(sumframe,7)*sum_step)+1:sample(sumframe+1,7)*sum_step,fluor)=sample(sumframe+1,2);

            y_pos((sample(sumframe,7)*sum_step)+1:sample(sumframe+1,7)*sum_step,fluor)=sample(sumframe+1,4);
        end
    end

    x_pos((sample(end,7)*sum_step)+1:end,fluor)=sample(end,2);
    y_pos((sample(end,7)*sum_step)+1:end,fluor)=sample(end,4);

    for sumframe=1:size(sample,1)
        sumframe_no=sample(sumframe,7);
        fluor_det(((sumframe_no-1)*sum_step)+1:sumframe_no*sum_step,fluor)=1;
    end
end

x_pos_round=round(x_pos);
y_pos_round=round(y_pos);

for frame=1:nfiles

    filepath=filelist{frame};
    image = double(imread(char(filepath)));

    for fluorno=1:nfluors
        left=x_pos_round(frame,fluorno)-boxsize;

```

```

right=x_pos_round(frame,fluorno)+boxsize;
up=y_pos_round(frame,fluorno)-boxsize;
down=y_pos_round(frame,fluorno)+boxsize;
%disp([fluorno, up, down, left, right])
extract=image(up:down,left:right);
if sub_back==1
up2=up-1; down2=down+1; left2=left-1; right2=right+1;
side=2*boxsize+3;
background_inds=zeros(4*side-4,2);
background_inds(1:side,1)=up2;
background_inds(1:side,2)=left2:right2;
background_inds(side+1:2*side-1,1)=up2+1:down2;
background_inds(side+1:2*side-1,2)=right2;
background_inds(2*side:3*side-2,1)=down2;
background_inds(2*side:3*side-2,2)=right2-1:-1:left2;
background_inds(3*side-1:end,1)=down2-1:-1:up2+1;
background_inds(3*side-1:end,2)=left2;
background=zeros(4*side-4,1);
% Check background inds are within image bounds
[row,~]=find(background_inds<1);
if ~isempty(row)
    background_inds(row,:)=[];
    background(length(background)-length(row)+1:end,:)=[];
end
%disp(background_inds)
for dum=1:length(background)

background(dum)=image(background_inds(dum,1),background_inds(dum,2)
);

    end
    background=mean(background);
    intensities(frame,fluorno,1)=sum(sum(extract))-
(background*((2*boxsize+1)^2));
    else
        intensities(frame,fluorno,1)=sum(sum(extract));
    end
end
end
end
end

```

CKfilteredit3

This function reproduces the forward-backward non-linear filter described by Chung and Kennedy[77].

Inputs

data: m by n matrix containing intensity values for all n fluorophores in all m frames, created by the function `extract_int_local`

predictors: row vector containing averaging window widths

width: analysis window size over which to calculate forward and backward weights

p: factor determining relative weights to assign each forward and backward predictor

predictpref: optional argument, row vector included to alter weighting given to different predictors

Outputs

newdata: m by n matrix containing smoothed intensity traces for all n fluorophores in all m frames

```
function [ newdata ] = CKfilteredit3( data, predictors, width, p,
predictpref )
%UNTITLED3 Summary of this function goes here
%   Detailed explanation goes here

% data = noisy signal in column vector form, different fluors in
different
% columns
% predictors = row vector containing averaging window widths
% width = analysis window size over which to calculate forward and
backward
%           weights; typically 4<=width<=20
% p = factor (0<p<101) determining relative weights to assign each
forward and
%       backward predictor; small p gives increased noise reduction,
large p
%       better preserves signal amplitude and edges
% predictpref (optional variable) = row vector containing
constants
%                                     to indicate which predictors
should be                             preferred on average. If left
%                                     empty,
%                                     all predictors are given equal
%                                     weighting
% length(predictpref)=length(predictors)
%                                     sum(predictpref)=0.5

clean=1;
[A B]=size(data);
predictor_no=length(predictors);
XF=zeros(A,predictor_no);
XB=zeros(A,predictor_no);
FK=zeros(A,predictor_no);
BK=zeros(A,predictor_no);
BKraw=zeros(A,predictor_no);
FKraw=zeros(A,predictor_no);
BKrawsum=zeros(A,1);
FKrawsum=zeros(A,1);
newdata=zeros(A,B);

%all predictors are given equal weighting if no values entered for
%predictpref
if nargin<5
predictpref=zeros(predictor_no);
predictpref(1:predictor_no)=1/(2*predictor_no);
end

for fluor=1:B
% form an average from the previous data points

for window=1:predictor_no
for k=predictors(window)+1:A
XF(k,window)=mean(data((k-predictors(window)):(k-1),fluor));
end
end
end
```

```

end

% form an average from the upcoming data points

for window=1:predictor_no
for k=1:A-predictors(window)
    XB(k,window)=mean(data((k+1):(k+predictors(window)),fluor));
end
end

% calculate the coeff

for window=1:predictor_no
for k=width+1:A
    total=sum((data((k-width+1):k,fluor)-XF((k-
width+1):k,window)).^2);
    FKraw(k,window)=predictpref(window)*(total^(-p));
end
end

for window=1:predictor_no
for k=1:A-width
    total=sum((data(k:(k+width-1),fluor)-XB(k:(k+width-
1),window)).^2);
    BKraw(k,window)=predictpref(window)*(total^(-p));
end
end

%normalise forward and backward weights

for k=1:A
    FKrawsum(k)=sum(FKraw(k,:));
    BKrawsum(k)=sum(BKraw(k,:));
end

for window=1:predictor_no
BK(:,window)=BKraw(:,window)./(BKrawsum+FKrawsum);
FK(:,window)=FKraw(:,window)./(BKrawsum+FKrawsum);
end

%calculate the smoothed estimate

temp=(FK.*XF)+(BK.*XB);
for k=1:A
    newdata(k,fluor)=sum(temp(k,:));
end
end

%{
clf;
plot(newdata(clean:end),'b. ');
hold on;
plot(data(clean:end),'r');
%}

end

```

fillNaNs

Inputs

CK_int: matrix of smoothed intensity traces created by the function CKfilteredit3

Outputs

CK_int: matrix of smoothed intensity traces after checking for and correcting any NaN values

```
function [ CK_int ] = fillNaNs( CK_int )
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
% Finds NaNs which might occur in CK traces after several
iterations and
% replaces them with the preceeding value(s)

nans=isnan(CK_int);
[row,col]=find(nans);

if isempty(row)
    disp('No NaNs found')
    return
else
    for nan=1:length(row)
        CK_int(row(nan),col(nan))=CK_int(row(nan)-1,col(nan));
    end
end

end
```

pairwise_diffs

Inputs

filtered_data: matrix of smoothed intensity traces after checking for and correcting any NaN values

Outputs

pair_diffs: matrix containing pairwise differences of the intensity traces for every fluorophore

```
function [ pair_diffs ] = pairwise_diffs( filtered_data )
%UNTITLED Summary of this function goes here
%Calculates the pairwise differences in a CK filtered intensity
trace.

% Detailed explanation goes here
% filtered_data = filtered intensity traces in column vector
form (each
% fluorophore in a separate column)

[A B]=size(filtered_data);
pair_diffs=zeros(A*(A-1)/2,B);

for fluor=1:B
    dum=1;
```

```

        for time=1:A-1
            for next=1:A-time
                pair_diffs(dum,fluor)=filtered_data(time+next,fluor)-
filtered_data(time,fluor);
                dum=dum+1;
            end
        end
    end
end
end

```

makePDDF

Inputs

PDs: matrix containing pairwise differences of the intensity traces for every fluorophore, created by the function `pairwise_diffs`

nbins: number of bins to use in creating PDDFs

Outputs

freq: nbins by n matrix containing frequency counts divided across nbins bins for all n PDDFs

xout: nbins by n matrix containing nbins bin centres for all n PDDFs

```

function [ freq, xout ] = makePDDF( PDs, nbins )
%UNTITLED Summary of this function goes here
% Detailed explanation goes here
% Sort pairwise differences into pairwise difference distribution
functions
% for each fluorophore individually, in histograms with nbins
number of
% bins

nfluors=size(PDs,2);
freq=zeros(nbins,nfluors);
xout=zeros(nbins,nfluors);

for fluor=1:nfluors
    [freq(:,fluor),xout(:,fluor)]=hist(PDs(:,fluor),nbins);
end

end

```

PeakFinder

This function adapts and expands upon a peak flagging routine written by Nigel Reuel[76].

Inputs

intensities: m by n matrix containing intensity values for all n fluorophores in all m frames, created by the function `extract_int_local`

SG_freq: matrix containing smoothed frequency counts of PDDFs

freq: matrix containing raw frequency counts of PDDFs, created by the function `makePDDF`

xout: matrix containing bin centres of PDDFs, created by the function `makePDDF`

avg_window: window size for running average

graphics: indicates whether to display PDDF for each fluorophore showing identified peaks and valleys (set to 0 to suppress graphics or 1 to display graphics)

mole_factor: value between 0 and 1 determining the size of peaks deemed to be significant

SigD: value specifying the minimum distance allowed between a peak and its neighbouring valley for the peak to be deemed significant

SigBinfract: value between 0 and 1 specifying the minimum distance allowed between two adjacent peaks for them to remain separate

Outputs

peak_no: vector containing the number of identified peaks for each fluorophore

Peakfits: matrix containing information on all identified peaks (see below for key to columns)

noisepeak: matrix containing information on the zero-centred peak for each fluorophore (column contents are the same as for the Peakfits matrix)

Key to columns in Peakfits matrix

- 1: fluorophore number within current set
- 2: peak position (step intensity value)
- 3: peak height (frequency count)
- 4: standard deviation of Gaussian fit to peak

Custom functions called by PeakFinder

Gaussian

```
function [ peak_no, Peakfits, noisepeak ] = PeakFinder(
intensities, SG_freq, freq, xout, avg_window, graphics,
mole_factor, SigD, SigBinfract )
%UNTITLED3 Summary of this function goes here
% Detailed explanation goes here
% Coded by Nigel Reuel on 9.15.2010 updated in March 2011 by NFR
% (from NoRSE - Noise Reduction and State Evaluator, available at
% http://web.mit.edu/stranogroup/NoRSE.txt)

% Adapted by D Hesson Sept 2014
% This function finds the peaks of a histogram using a modified
running
% average algorithm + peak flagging routine

% avg_window = size of window to calculate running average over
(increasing
% this produces more of a rough approximation of peaks). Suggested
starting
% value = 3.

% graphics - 1 to display graphs for each fluorophore of identified
peaks
% and valleys; 0 to suppress graphs

% mole_factor = fraction determining the size of peaks deemed to
be
% significant e.g. mole_factor=0.5, peaks with heights< 1/2 of the
```

```

avg
% response are ignored. 0<=mole_factor<=1 (MoleL =
% (nframes/nBins/avg_window)*mole_factor). If not specified
% (mole_factor=[]), mole_factor is set to 0.5

% SigD = significant distance between identified peak and its
neighbouring
% valleys - if the calculated distance is smaller than the
threshold
% specified, the peak is combined with its neighbour. If SigD is
not
% specified (SigD=[]), it is set equal to MoleL (the factor used
to reject small
% peaks)

% SigBinfract = percentage determining how close peaks can be to
remain
% unique (SigBinD = nBins*SigBinfract). If not specified (=[]),
% SigBinfract is set to 0.025. Decreasing the multiplied factor
(0.025)
% will allow for peaks closer together to remain unique,
increasing the
% factor causes more of the flagged peaks to be unified into
single
% 'significant' peaks.
%
nframes=sum(freq(:,1));
SG_freq=SG_freq';
freq=freq';
xout=xout';
[nfluors,nBins] = size(SG_freq);
peak_no=zeros(nfluors,1);
noisepeak=zeros(nfluors,4);
noisepeak(:,1)=(1:nfluors)';
%
% Centered running average algorithm to help find peaks:
%
T_ra = zeros(nfluors,nBins);
for fluor = 1:nfluors
    for bin = 1:nBins
        if bin < (avg_window + 1)
            T_ra(fluor,bin) =
sum(SG_freq(fluor,1:bin+avg_window))/(bin+avg_window);
        elseif bin >= (avg_window+1) && bin <= (nBins-avg_window)
            T_ra(fluor,bin) = sum(SG_freq(fluor,bin-
avg_window:bin+avg_window))/(avg_window*2+1);
        else
            T_ra(fluor,bin) =
sum(SG_freq(fluor,bin:nBins))/(nBins-bin+1);
        end
    end
end
%xlswrite('T_ra',T_ra);
%
%-----Peak finding algorithm-----
%
%
% Base function on physical example of climbing up and down hills.
Drop

```



```

% flags on supposed peaks. Eliminate "peaks" that are molehills.
Combine
% peaks that are close neighbors.
Flags = zeros(nfluors,nBins,6);
% Layer 1 = Peak Flag
% Layer 2 = Right Valley Bin#
% Layer 3 = Left Valley Bin#
% Layer 4 = Peak magnitude (Histogram Count)
% Layer 5 = Right Valley magnitude (Histogram Count)
% Layer 6 = Left Valley magnitude (Histogram Count)
% Molehill level (number of responses that are not significant -
some fraction of the avg response)
%
%
%
if isempty(mole_factor)
    MoleL = (nframes/nBins/avg_window)*0.5;
else
    MoleL = (nframes/nBins/avg_window)*mole_factor; % =
#responses/#bins/SW ~= 1/2 of the Avg. Response - no significance.
end
% The multiplied factor ('0.5') can be decreased to allow smaller
peak heights to be deemed 'significant'
% or increased to allow for only the highest peaks to be flagged.
%

Peakfits = zeros(1,4);
for fluor = 1:nfluors
    for bin = 1:nBins
        if T_ra(fluor,bin) > MoleL
            % Logic to find significant peaks at endpoints (and the
R/L
            % valleys)
            if bin == 1 && T_ra(fluor,bin) > T_ra(fluor,bin+1)
                Flags(fluor,bin,1) = 1;
                Flags(fluor,bin,4) = T_ra(fluor,bin);
                Pval = T_ra(fluor,bin);
                Pnext = T_ra(fluor,bin+1);
                index = bin;
                % Find right hand valley:
                while Pval > Pnext
                    index = index + 1;
                    Pval = T_ra(fluor,index);
                    Pnext = T_ra(fluor,index+1);
                end
                Flags(fluor,bin,2) = index;
                Flags(fluor,bin,5) = T_ra(fluor,index);
            elseif bin == 1 && T_ra(fluor,bin) <= T_ra(fluor,bin+1)
                Flags(fluor,bin,1) = 0;
            elseif bin == nBins && T_ra(fluor,bin) > T_ra(fluor,bin-
1)
                Flags(fluor,bin,1) = 1;
                Flags(fluor,bin,4) = T_ra(fluor,bin);
                Pval = T_ra(fluor,bin);
                Pprior = T_ra(fluor,bin-1);
                index = bin;
                % Find left hand valley:
                while Pval > Pprior
                    index = index - 1;

```

```

        Pval = T_ra(fluor,index);
        Pprior = T_ra(fluor,index-1);
    end
    Flags(fluor,bin,3) = index;
    Flags(fluor,bin,6) = T_ra(fluor,index);
    elseif bin == nBins && T_ra(fluor,bin) <=
T_ra(fluor,bin-1)
        Flags(fluor,bin,1) = 0;
        % Logic to find peaks at all midpoints (and their R/L
valleys)
        elseif T_ra(fluor,bin) >= T_ra(fluor,bin-1) &&
T_ra(fluor,bin) >= T_ra(fluor,bin+1)
            Flags(fluor,bin,1) = 1;
            Flags(fluor,bin,4) = T_ra(fluor,bin);
            % Find left hand valley:
            Pval = T_ra(fluor,bin);
            Pprior = T_ra(fluor,bin-1);
            index = bin;
            if bin > 2
                Switch = 1;
            else
                Switch = 0;
                index = 1;
            end
            while Pval > Pprior && Switch == 1;
                index = index - 1;
                Pval = T_ra(fluor,index);
                Pprior = T_ra(fluor,index-1);
                if index == 2
                    Switch = 0;
                    index = 1;
                end
            end
            Flags(fluor,bin,3) = index;
            Flags(fluor,bin,6) = T_ra(fluor,index);
            % Find right hand valley:
            Pval = T_ra(fluor,bin);
            Pnext = T_ra(fluor,bin+1);
            index = bin;
            if bin < nBins - 1
                Switch = 1;
            else
                Switch = 0;
                index = nBins;
            end
            while Pval > Pnext && Switch == 1;
                index = index + 1;
                Pval = T_ra(fluor,index);
                Pnext = T_ra(fluor,index+1);
                if index == nBins - 1;
                    Switch = 0;
                    index = nBins;
                end
            end
            Flags(fluor,bin,2) = index;
            Flags(fluor,bin,5) = T_ra(fluor,index);
        end
    end
end
end
%{

```

```

X = (1:200)';
Y1 = Flags(i,:,1)';
Y2 = T_ra(i,:);
plot(X,Y1,X,Y2)
%}

[~,PeakVec]=find(Flags(fluor,:,1)); %PeakVec contains the bin
numbers of identified peaks
% Now analyze peak pairs:
% Specify significant peak valley distance and significant
bin
% distance:
%
%
if isempty(SigD)
SigD = MoleL;
end
%
%
if isempty(SigBinfract)
SigBinD = nBins*(.025); % = 2.5% of the total bin #.
else
SigBinD = nBins*SigBinfract;
end
% Decreasing the multiplied factor (0.025) will allow for
peaks closer
% together to remain unique, increasing the factor causes
more of the
% flagged peaks to be unified into single 'significant'
peaks.
%
%
Npeaks = length(PeakVec);
% Determine the peak pairwise interactions...each will have
a right
% hand value and left hand value (except for the endpoints):
% 1: Unique
% 2: Combine
% 3: Ignore
% Create a matrix to recieve these calculations:
PeakCode = zeros(2,Npeaks); % Row1 = to right, Row2 = to
left
for peak = 1:Npeaks-1
% Determine the peak bin numbers:
PBN1 = PeakVec(1,peak);
PBN2 = PeakVec(1,peak+1);
% Calculate the valley distances (left to right)
VD1 = abs(Flags(fluor,PBN1,4) - Flags(fluor,PBN1,5));
VD2 = abs(Flags(fluor,PBN2,6) - Flags(fluor,PBN2,4));
%
if VD1 <= SigD && VD2 <= SigD
% Combine
PeakCode(1,peak) = 2;
PeakCode(2,peak+1) = 2;
elseif VD1 >= SigD && VD2 <= SigD
% Left Peak sig, right peak ignore
PeakCode(1,peak) = 1;
PeakCode(2,peak+1) = 3;
elseif VD1 <= SigD && VD2 >= SigD

```

```

        % Left peak ignore, right peak significant
        PeakCode(1,peak) = 3;
        PeakCode(2,peak+1) = 1;
    elseif VD1 >= SigD && VD2 >= SigD
        % Both peaks are significant
        PeakCode(1,peak) = 1;
        PeakCode(2,peak+1) = 1;
    end
end
% Logic for the peak endpoints
Left = abs(Flags(fluor,PeakVec(1,1),4) -
Flags(fluor,PeakVec(1,1),6));
if Left >= SigD
    PeakCode(2,1) = 1;
else
    PeakCode(2,1) = 3;
end
Right = abs(Flags(fluor,PeakVec(1,Npeaks),4) -
Flags(fluor,PeakVec(1,Npeaks),5));
if Right >= SigD
    PeakCode(1,Npeaks) = 1;
else
    PeakCode(1,Npeaks) = 3;
end
% Now determine what to do with the peaks. Look at
combinations
% reading left to right. Put significant peaks into a
vector.
SPV = 0;
dum = 1;
count = 1;
ignore = 0;
while dum <= Npeaks
    C1 = PeakCode(1,dum);
    C2 = PeakCode(2,dum);
    if C1 == 1 && C2 == 1
        % This is a bonafide unique peak. Put it in the
vector.
        SPV(count,1) = PeakVec(1,dum);
        SPV(count,2) = Flags(fluor,PeakVec(1,dum),2);

        if ignore==0
            SPV(count,3) = Flags(fluor,PeakVec(1,dum),3);
        elseif Flags(fluor,PeakVec(1,dum-ignore),3)==0
            SPV(count,3) = Flags(fluor,PeakVec(1,dum),3);
        else
            SPV(count,3) = Flags(fluor,PeakVec(1,dum-
ignore),3);
        end
        count = count + 1;
        dum = dum+1;
        ignore = 0;
    elseif C1 == 2
        % This peak will be merged with other peak(s)
        count2 = 2;
        Merge = PeakVec(1,dum);
        while C1 == 2
            dum = dum + 1;
            C1 = PeakCode(1,dum);

```

```

        Merge(1,count2) = PeakVec(1,dum);
        count2 = count2 + 1;
    end
    % Insert logic here to eliminate valleys and slow
    rises (see
        % what is happening at the significant enpoints!)
        End1 = Flags(fluor,Merge(1,1),4) -
Flags(fluor,Merge(1,1),6);
        End2 = Flags(fluor,Merge(1,end),4) -
Flags(fluor,Merge(1,end),5);
        Distance = Merge(1,end) - Merge(1,1);
        if End1 >= SigD && End2 >= SigD
            SPV(count,1) = round(mean(Merge));
            SPV(count,2) = Flags(fluor,Merge(1,end),2);
            SPV(count,3) = Flags(fluor,Merge(1,1),3);
            count = count + 1;
            dum = dum + 1;
        elseif End1 >= SigD && End2 <= (SigD)*(-1) &&
Distance >= SigBinD
            SPV(count,1) = round(mean(Merge));
            SPV(count,2) = Flags(fluor,Merge(1,end),2);
            SPV(count,3) = Flags(fluor,Merge(1,1),3);
            count = count + 1;
            dum = dum + 1;
        elseif End1 <= SigD*(-1) && End2 >= (SigD) &&
Distance >= SigBinD
            SPV(count,1) = round(mean(Merge));
            SPV(count,2) = Flags(fluor,Merge(1,end),2);
            SPV(count,3) = Flags(fluor,Merge(1,1),3);
            count = count + 1;
            dum = dum + 1;
        end
    else % Represents 1/3, 3/1, and 3/3 combos which are all
    ignored!!
        dum = dum + 1;
        ignore = ignore + 1;
    end
end

Nsigpeaks = size(SPV,1);
if Nsigpeaks>0
    %Discard zero peak and positive peaks
    zero_peakid=dsearchn(xout(fluor,SPV(:,1))',0);
    noisepeakSPV=SPV(zero_peakid,:);
    SPV(zero_peakid:end,:)=[];
    %Discard peaks too close to either end
    [row,~]=find(SPV==0);
    if ~isempty(row)
        SPV(row,:)=[];
    end
end

Nsigpeaks=size(SPV,1);
%disp(fluor)

if graphics==1
    figure
    plot(xout(fluor,:),T_ra(fluor,:))
    hold on

```

```

scatter(xout(fluor,SPV(:,1)),T_ra(fluor,SPV(:,1)),'*','r')
    [~,~,v]=find(SPV(:,2));
    scatter(xout(fluor,v),T_ra(fluor,v),'*','k')
    [~,~,v]=find(SPV(:,3));
    scatter(xout(fluor,v),T_ra(fluor,v),'*','g')
    title(['Identified peaks for fluorophore ',
int2str(fluor)])
    legend('Smoothed PDDF','Identified peaks','Righthand
valleys','Lefthand valleys','Location','NorthWest')
    end

    % Fit raw PDDF with Gaussians, extracting peaks between
their
    % identified bounding valleys and using the identified peak
position as a
    % starting estimate

    % Fit the zero peak (to determine the noise)
x_ex = xout(fluor,noisepeakSPV(1,3):noisepeakSPV(1,2));
y_ex = T_ra(fluor,noisepeakSPV(1,3):noisepeakSPV(1,2));

    if isempty(x_ex)
        data=zeros(1,4);
        Peakfits = [Peakfits;data];
        break
    end

    %disp([x_ex, y_ex])
    coeff = zeros(4,1);
    coeff(1,1)=T_ra(fluor,noisepeakSPV(1,1));
    coeff(2,1)=xout(fluor,noisepeakSPV(1,1));
    coeff(3,1)=abs(x_ex(1,1)-x_ex(end,1))/4;
    coeff(4,1)=min(y_ex);
    %disp(coeff)
    options=optimset('Display','off');
    %options=optimset('MaxFunEvals',1e6);

fit_coeffs=lsqcurvefit(@Gaussian,coeff,x_ex,y_ex,[],[],options);
    fit = fit_coeffs(1,1)*exp(-1*((x_ex-
fit_coeffs(2,1)).^2/(2*(fit_coeffs(3,1)).^2)))+fit_coeffs(4,1);

    noisepeak(fluor,2) = fit_coeffs(2,1);
    noisepeak(fluor,3) = fit_coeffs(1,1) + fit_coeffs(4,1);
    noisepeak(fluor,4) = fit_coeffs(3,1);

    if graphics==1
        figure
        plot(xout(fluor,:),freq(fluor,:))
        title(['Fitted peaks for fluorophore ',
int2str(fluor)];'Green is accepted, red is discarded, black is
noise'})
        hold on
        plot(x_ex,fit,'k')
    end

    data=zeros(Nsigpeaks,4);

```

```

data(:,1)=fluor;

% Fit the identified peaks
for peak = 1:Nsigpeaks
    x_ex = xout(fluor,SPV(peak,3):SPV(peak,2))';
    y_ex = freq(fluor,SPV(peak,3):SPV(peak,2))';
    coeff = zeros(4,1);
    coeff(1,1)=T_ra(fluor,SPV(peak,1));
    coeff(2,1)=xout(fluor,SPV(peak,1));
    coeff(3,1)=abs(x_ex(1,1)-x_ex(end,1))/4;
    coeff(4,1)=min(y_ex);
    options=optimset('Display','off');

    fit_coeffs=lsqcurvefit(@Gaussian,coeff,x_ex,y_ex,[],[],options);
    fit = fit_coeffs(1,1)*exp(-1*((x_ex-
    fit_coeffs(2,1)).^2/(2*(fit_coeffs(3,1)).^2)))+fit_coeffs(4,1);

    data(peak,2) = fit_coeffs(2,1);
    data(peak,3) = fit_coeffs(1,1) + fit_coeffs(4,1);
    data(peak,4) = fit_coeffs(3,1);

    % Check the Gaussian fit is reasonable; if not, discard
the peak
    if (fit_coeffs(2,1)<x_ex(end,1) &&
    fit_coeffs(2,1)>x_ex(1,1) ...
        && abs(fit_coeffs(3,1))<=abs(x_ex(end,1) -
x_ex(1,1)))
        if graphics==1
            plot(x_ex,fit,'g')
        end
    else
        data(peak,:)=0;
        if graphics==1
            plot(x_ex,fit,'r')
        end
    end
end

Peakfits = [Peakfits;data];

end

discard= Peakfits(:,1)==0;
Peakfits(discard,:)=[];

% Check that all identified peaks are sensible i.e. not just
sustained
% noise fluctuations
int_sample=zeros(1,3);
npoints=size(intensities,1);
for fluor = 1:nfluors
    int_sample(1,1)=mean(intensities(1:5,fluor));
    int_sample(1,2)=mean(intensities(npoints-
avg_window:end,fluor));
    int_sample(1,3)=abs(int_sample(1,1)-int_sample(1,2));

    rows=find(Peakfits(:,1)==fluor);
    peak_no(fluor,1)=length(rows);

```

```

    if ~isempty(rows)
        peakvals=Peakfits(rows,2);
        peaksizes=Peakfits(rows,3);
        total=abs(sum(peakvals));
        total_uncert=2.4*sqrt(sum(Peakfits(rows,4).^2));
        noisesig=noisepeak(fluor,4);

        if ((int_sample(1,3)-4.8*noisesig>total+total_uncert)
&&...
            (total<2.4*abs(noisesig)+abs(noisepeak(fluor,2))))
            if length(rows)>1
                Peakfits(rows(2:end,1),:)=[];
            end
            peak_no(fluor,1)=1;
        elseif (int_sample(1,3)-4.8*noisesig>total+total_uncert)
            Peakfits(end+1,:)=zeros(1,4);
            Peakfits(end,1)=fluor;
            Peakfits(end,2)=-1*int_sample(1,1);
            Peakfits(end,3:4)=NaN;
            peak_no(fluor,1)=peak_no(fluor,1)+1;
        elseif (int_sample(1,3)<(max(abs(peakvals))-
4.8*abs(noisesig)) || ...
            int_sample(1,1)<0 )
            Peakfits(rows,:)=[];
            peak_no(fluor,1)=0;
        elseif
(min(abs(peakvals))<(2.4*abs(noisesig)+abs(noisepeak(fluor,2))))
&&...
            peaksizes(end,1)>0.95*noisepeak(fluor,3))
            diffs=abs(diff(peakvals));
            if isempty(diffs)
                Peakfits(rows,:)=[];
                peak_no(fluor,1)=0;
            elseif length(diffs)==1
                Peakfits(rows(end,1),:)=[];
                peak_no(fluor,1)=1;
            else
                diffs(end,:)=[];
                %disp(diffs)
                Peakfits(rows(end,1),:)=[];
                merge1=find(diffs<min(abs(peakvals))+noisesig);
                merge2=find(diffs>min(abs(peakvals))-noisesig);
                merge=intersect(merge1,merge2);
                %disp('Peakfits size')
                %disp(size(Peakfits(rows(1:end-1,:),:),1))
                %disp(merge)
                if ~isempty(merge)
                    for dum=merge'

Peakfits(rows(dum,1),2)=mean([Peakfits(rows(dum,1),2),Peakfits(row
s(dum+1,1),2)]);

                                Peakfits(rows(dum,1),3)=NaN;
                                Peakfits(rows(dum,1),4)=NaN;

                                end
                                Peakfits(rows(merge+1,1),:)=[];
                                peak_no(fluor,1)=length(rows)-length(merge)-1;
                                end
                            end
                        end
                    end
                end
            end
        end
    end
end

```



```

        end
    end

end

```

log_background_removal

This function is based on code written by Robert Turner (MBB, The University of Sheffield).

Custom functions called by log_background_removal

LM_oslash (see Appendix A:)

```

function [ filteredimage, fits_overlay ] = log_background_removal(
filelist, reconset, imageno )
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
% Removes background from image number 'imageno' in a series of
images
% using a Laplacian of Gaussian filter. Need to specify
% reconset parameters by altering them within 'LM_MakeReconset'
and running
% the function.

nfiles=size(filelist,1);

%generate minimum image used for subtraction
if (reconset.subtractmin==1)
    j=0;
    for i=1:round(nfiles/100):nfiles
        j=j+1;
        filepath=filelist{i};
        images(j, :, :) = double(imread(char(filepath)));
    end
    minimage=squeeze(min(images));
else
    %open up an image just to check dimensions
    images(1, :, :) = double(imread(char(filelist{1})));
end

dimension_x=size(images,2);
dimension_y=size(images,3);

myxlim=[0 dimension_x];
myylim=[0 dimension_y];

%slash direction
slash=LM_oslash;

filepath=filelist{imageno};
[folder, filenameonly, ~]=fileparts(filepath);

%get framenummer
framenummer=str2double(regexp(filenameonly, '[0-9]+', 'match'));
rawimage=double(imread(char(filepath)));

if reconset.filter==1

```

```

if reconset.subtractmin==1
    %subtract minimum values
    filteredimage=rawimage-minimage;
else
    filteredimage=rawimage;
end

%apply laplacian of gaussian filter
LOG = fspecial('log',[10 10],reconset.log_width);
filteredimage = -1*imfilter(filteredimage,LOG,'replicate');
activeimage=filteredimage;
else
    activeimage=rawimage;
end

figure;
imagesc(rawimage);
set(gca,'DataAspectRatio',[1 1 1]);
set(gca,'YDir','reverse')
title(strcat('Raw File:
',int2str(imageno),'/',int2str(nfiles)),'FontWeight','bold');
colormap('gray');

fits_overlay=figure;
imagesc(filteredimage);
set(gca,'DataAspectRatio',[1 1 1]);
set(gca,'YDir','reverse')
title(strcat('Filtered File:
',int2str(imageno),'/',int2str(nfiles)),'FontWeight','bold');
colormap('gray');

end

```

Gaussian

```

function F = Gaussian(c,xdata)

F = c(1)*exp(-1*((xdata-c(2)).^2/(2*(c(3)).^2)))+c(4);

```

Appendix C: MATLAB code for analysis of fluorescent nucleotide binding

A list of the arguments is provided with explanations for each user-manipulated function. See Section 4.3. for an explanation of their implementation. Functions called by functions but not requiring user input are given at the end without explanatory notes.

check_sep_allframes

Inputs

reconset: struct containing the fields and values specified in LM_MakeReconset (see Appendix A:)

events: matrix containing information on the fluorophore localizations (see LM_reconstruct in Appendix A: for the column key)

filelist: cell array holding the file path to each image file in the series, created by the function LM_filelist (see Appendix A:)

Outputs

events_new: matrix containing information on the fluorophore localizations, excluding those fluorophores deemed as overlapping (see LM_reconstruct for the column key)

avg_xy: matrix containing fluorophore identifier numbers and average positions in column format for refined fluorophore list (column 1: identifier number; column 2: average x-coordinate; column 3: average y-coordinate)

Custom functions called by check_sep_allframes

log_background_removal (see Appendix B:)

```
function [ events_new, avg_xy ] = check_sep_allframes( reconset,
events, filelist )
%UNTITLED Summary of this function goes here
% Detailed explanation goes here
% Removes fluorophores occurring within the separation distance of
%each other. Assumes localisations have been performed in multiple
frames,
%and uses an average of the resulting xy positions to eliminate
spatially overlapping
%fluorophores occurring at any time (i.e. not just those occurring
in the
%same frame). Significant drift in the sample might cause
problems.
%
% Choose to plot retained and discarded localisations by setting
% reconset.graphics=1.

separation=reconset.spacing;
fluor_nos=unique(events(:,10));
nfluors=length(fluor_nos);
avg_xy=zeros(3,nfluors);
fluor_nos=sort(fluor_nos);
events_new=events;

dum=1;
for fluor=fluor_nos'
    rows=find(events(:,10)==fluor);
    xvals=events(rows,2);
```

```

        yvals=events(rows,4);
        avg_xy(1,dum)=fluor;
        avg_xy(2,dum)=mean(xvals);
        avg_xy(3,dum)=mean(yvals);
        dum=dum+1;
    end

    close=0;
    for fluor=1:nfluors
        ind_samex1=find(avg_xy(2,:)>(avg_xy(2,fluor)-separation));
        ind_samex2=find(avg_xy(2,:)<(avg_xy(2,fluor)+separation));
        ind_samex=intersect(ind_samex1,ind_samex2);
        omit= ind_samex==fluor;
        ind_samex(omit)=[];

        if ~isempty(ind_samex)
            for dum=ind_samex
                %disp(dum)
                if ((avg_xy(3,dum)>(avg_xy(3,fluor)-separation)) &&
                    (avg_xy(3,dum)<(avg_xy(3,fluor)+separation)))
                    close=[close;dum];
                end
            end
        end
    end
end

% Remove starting zero and repeated index values
close(1)=[];
close=unique(close);
close_fluors=avg_xy(1,close');
avg_xy(:,close')=[];

if ~isempty(close_fluors)
    del_rows=0;
    xy_find=0;
    for dum=close_fluors
        del=find(events(:,10)==dum);
        xy=find(events(:,10)==dum,1,'first');
        xy_find=[xy_find;xy];
        del_rows=[del_rows;del];
    end
    del_rows(1)=[];
    xy_find(1)=[];
    del_xy=[events(xy_find,2),events(xy_find,4)];
    events_new(del_rows,:)=[];
    else del_xy=0;
end

if reconset.graphics==1
    [~, keep_rows, ~]=unique(events_new(:,10),'first');
    keep_xy=[events_new(keep_rows,2),events_new(keep_rows,4)];
    [ignore, fits_overlay] = log_background_removal( filelist,
reconset, 1 );
    figure(fits_overlay)
    hold on;
    scatter(keep_xy(:,1),keep_xy(:,2),[],'b')
    if del_xy~=0
        scatter(del_xy(:,1),del_xy(:,2),[],'r')
    end
end

```

```
        hold off;  
end  
  
end
```

extract_int_local

See Appendix B: for the code of extract_int_local.

Fluorophore structure fields

fluorno: fluorophore identifier number as assigned by LM_emitters (see Appendix A:)

avgxy100fr: matrix containing the x- and y-coordinates extracted from an image series of averaged frames, as created by extract_int_local (see Appendix B:)

det100fr: vector of ones and zeros indicating whether a fluorophore was detected by the function LM_reconstruct (see Appendix A:) in each frame (1: fluorophore detected; 0: fluorophore not detected)

xyeachframe: matrix containing the x- and y-coordinates, amplitude, and x- and y-widths from Gaussian fits of fluorophores in those single frames of the original image series where they were detected by the function LM_reconstruct (indicated by det100fr)

int: vector containing the intensity value as extracted by extract_int_local for every frame of the original image series

CKint: intensity trace smoothed using a Chung Kennedy filter

det_thresh: logical vector of ones and zeros indicating whether a fluorophore was on or off in each frame as determined by applying a threshold to the smoothed intensity trace CKint

CKintcorr: smoothed intensity trace after subtraction of an exponential baseline

intcorr: raw intensity trace after subtraction of the same exponential baseline

bkgdsub: value indicating whether baseline subtraction was possible or successful (1) or not (0)

avgCKamp_bkgd: matrix containing the average background value in the raw and smoothed intensity traces along with its standard deviation before and after baseline subtraction

onofftimes_fr: matrix containing information on events identified in the CKintcorr intensity trace (see below for the key to the rows)

false_rate: the average number of false events detected per frame during threshold-crossing analysis

onofftimes_res: matrix containing information on events after imposition of a resolution limit (see below for the key to the rows)

Key to rows in onofftimes fr matrix

- 1: Event number
- 2: Value indicating 'on' (1) or 'off' (0)
- 3: Uncorrected event duration (frames)
- 4: Event duration corrected for short events (frames)
- 5: Event start time (frames)
- 6: Event end time (frames)
- 7: Event amplitude

Key to rows in onofftimes res matrix

- 1: Original event number
- 2: Event number after concatenation
- 3: Value indicating 'on' (1) or 'off' (0)
- 4: Event start time (frames)
- 5: Event end time (frames)
- 6: Event start time (time)
- 7: Event end time (time)
- 8: Event duration (time)
- 9: Event amplitude

CKfilt_struct

Inputs

fluorstruct: array of structures containing details of fluorophore attributes (see Fluorophore structure fields)

predictors: row vector containing averaging window widths

width: analysis window size over which to calculate forward and backward weights

p: factor determining relative weights to assign each forward and backward predictor

predictpref: optional argument, row vector included to alter weighting given to different predictors

Outputs

fluorstruct: array of structures containing details of fluorophore attributes, including the newly calculated CKint field (see Fluorophore structure fields)

Custom functions called by CKfilt_struct

CKfilteredit3 (see Appendix B:)

```

function [ fluorstruct ] = CKfilt_struct( fluorstruct, predictors,
width, p, predictpref )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here

[~, nfluors]=size(fluorstruct);

for fluor=1:nfluors
    data=fluorstruct(fluor).int';
    newdata=CKfilteredit3(data, predictors, width, p,
predictpref);
    fluorstruct(fluor).CKint=newdata';
    if find(isnan(newdata), 1)
        disp(['NaNs present in trace # ',int2str(fluor)])
    end
end

end

```

halfampthresh_multilevel

Inputs

fluorophores: array of structures containing details of fluorophore attributes (see Fluorophore structure fields)

A0: average signal amplitude above the background intensity level

backgd: average background intensity level

Tr: rise time of the Chung Kennedy filter used to smooth the raw intensity traces

params: parameter values used to correct for short event durations

fn: handle to the function (sqrtx) used to calculate corrections to short event durations

Outputs

fluorophores: array of structures containing details of fluorophore attributes, including the newly calculated det_thresh, CKintcorr, intcorr, bkgdsub, avgCKamp_bkgd, onofftimes_fr and falserate fields

Custom functions called by halfampthresh_multilevel

sqrtx

```

function [ fluorophores ] = halfampthresh_multilevel(
fluorophores, A0, backgd, Tr, params, fn )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
% Perform half amplitude threshold crossing analysis on CK
filtered
% intensity time traces to determine in which frames a fluorophore
is on or
% off. Correct short pulse durations to compensate for the
averaging effect
% of the filter and adjust all pulse durations accordingly.

% fluorophores - struct containing all information on fluorophores
of a
% data set
% A0 - smallest event amplitude determined from examination of

```

```

CKint traces,
% calculated as the difference between the background level and
fluorophore
% intensity
% backgd - background intensity level
% Tr - risetime of CK filter (i.e. longest short pulse duration
which must
% be corrected for due to the averaging effect of the filter)
% params - parameters to be passed to the function used to
calculate the
% short pulse correction
% fn - handle to the function which calculates the corrected short
pulse
% length

nfluors=size(fluorophores,2);
nframes=size(fluorophores(1).int,2);
falserates=zeros(1,nfluors);
falsen=1;

for fluor=1:nfluors

    %Determine rough on and off times
    roughon= fluorophores(fluor).CKint>((A0/2)+backgd);
    roffframes=find(roughon==0);
    noffframes=length(roffframes);

    %Alter CKint trace to subtract the background and correct any
drift
    %in the baseline
    fluorophores(fluor).CKintcorr=zeros(1,nframes);
    transitstart=find(diff(roughon));
    ntrans=length(transitstart);

    %Determine whether baseline subtraction is possible; if not,
return
    %zeros as CKintcorr, and return '0' for
fluorophores(fluor).bkgdsub
    if ntrans==0 || noffframes<100
        fluorophores(fluor).bkgdsub=0;
        falserates(1)=[];
        continue
    end
    fluorophores(fluor).bkgdsub=1;
    offsects=zeros(2,noffframes);
    rawoffsects=zeros(1,noffframes);

    %Extract the baseline from the 'off' portions
%of the trace along with frame numbers
    offsects(1,:)=roffframes;
    offsects(2,:)=fluorophores(fluor).CKint(roffframes);

    %Fit an exponential line to the extracted baseline
    [estimates]=fitline(offsects(1,:),offsects(2,:));

    %Subtract the fitted baseline from the CKint data to give a
%baseline-corrected intensity trace
    baseline=linefinal(1:nframes,estimates);
    fluorophores(fluor).CKintcorr=fluorophores(fluor).CKint-

```



```

baseline;
    fluorophores(fluor).intcorr=fluorophores(fluor).int-baseline;

    %Calculate average amplitudes and backgrounds for individual
    %fluorophores using the rough off sections. Corrected
background
    %is the average background after subtraction of the calculated
baseline.
    fluorophores(fluor).avgCKamp_bkgd=zeros(2,2);
    fluorophores(fluor).avgCKamp_bkgd(1,1)=mean(offsects(2,:));
    fluorophores(fluor).avgCKamp_bkgd(2,1)=std(offsects(2,:));
    fluorophores(fluor).avgraw_bkgd=zeros(2,1);

    fluorophores(fluor).avgraw_bkgd(1,1)=mean(fluorophores(fluor).int(
roffframes));

    fluorophores(fluor).avgraw_bkgd(2,1)=std(fluorophores(fluor).int(r
offframes));

    fluorophores(fluor).avgCKamp_bkgd(1,2)=mean(fluorophores(fluor).CKi
ntcorr(roffframes));

    fluorophores(fluor).avgCKamp_bkgd(2,2)=std(fluorophores(fluor).CKi
ntcorr(roffframes));

    %Check background subtraction was successful by checking the
new
    %average background. If this is not close to zero, set
bkgdsub=0 &
    %discard CKintcorr.
    if abs(fluorophores(fluor).avgCKamp_bkgd(1,2))>1
        fluorophores(fluor).bkgdsub=0;
        fluorophores(fluor).CKintcorr=0;
        fluorophores(fluor).intcorr=0;
        fluorophores(fluor).avgCKamp_bkgd(1:2,2)=0;
        falserates(1)=[];
        continue
    end

    if isempty(Tr)
        continue
    end

    %Re-evaluate on and off times from corrected CKints,
correcting for short pulses.

    fluorophores(fluor).det_thresh=fluorophores(fluor).CKintcorr>(A0/2
);

    transitstart=find(diff(fluorophores(fluor).det_thresh));
    nevents=length(transitstart)+1;
    fluorophores(fluor).onofftimes_fr=zeros(7,nevents);
    fluorophores(fluor).onofftimes_fr(1,:)=1:nevents;
    if fluorophores(fluor).det_thresh(1)==0
        for event=2:2:nevents
            fluorophores(fluor).onofftimes_fr(2,event)=1;
        end
    else
        for event=1:2:nevents

```

```

        fluorophores(fluor).onofftimes_fr(2,event)=1;
    end
end

%Determine uncorrected event start and end times, and
interpolate to
%determine more closely where the pulse crosses the threshold
value
%(recording interpolated event start and end times too
temporarily)
temponofftimes=zeros(11,nevents);

temponofftimes(1:2,:)=fluorophores(fluor).onofftimes_fr(1:2,:);

temponofftimes(4,1)=0;
temponofftimes(5,1)=transitstart(1);
pulseendamp1=fluorophores(fluor).CKintcorr(transitstart(1));
pulseendamp2=fluorophores(fluor).CKintcorr(transitstart(1)+1);
addon=abs(pulseendamp1-(A0/2))/abs(pulseendamp1-pulseendamp2);
temponofftimes(6,1)=0;
temponofftimes(7,1)=temponofftimes(5,1)+addon;
for event=2:(nevents-1)
    pulseend=transitstart(event);
    temponofftimes(4,event)=transitstart(event-1);
    temponofftimes(5,event)=pulseend;
    temponofftimes(6,event)=temponofftimes(7,event-1);
    pulseendamp1=fluorophores(fluor).CKintcorr(pulseend);
    pulseendamp2=fluorophores(fluor).CKintcorr(pulseend+1);
    addon=abs(pulseendamp1-(A0/2))/abs(pulseendamp1-
pulseendamp2);
    temponofftimes(7,event)=pulseend+addon;
end
temponofftimes(4,nevents)=transitstart(nevents-1);
temponofftimes(5,nevents)=nframes;
temponofftimes(6,nevents)=temponofftimes(7,nevents-1);
temponofftimes(7,nevents)=nframes;

%Calculate uncorrected pulse durations
temponofftimes(3,:)=temponofftimes(5,:)-temponofftimes(4,:);
fluorophores(fluor).onofftimes_fr(3,:)=temponofftimes(3,:);
temponofftimes(8,:)=temponofftimes(7,:)-temponofftimes(6,:);

%Determine which are short pulse durations
shorthevents=find(fluorophores(fluor).onofftimes_fr(3,:) <= Tr);

%Correct interpolated short event durations
temponofftimes(9:10,:)=temponofftimes(6:7,:);
lengthchange=zeros(1,nevents);

for event=shorthevents
    interlength=temponofftimes(8,event);
    corrlength=fn(interlength,params);
    lengthchange(event)=(corrlength-interlength)/2;
end
for event=1:(nevents-1)
    if lengthchange(event)==0
        endchange=-lengthchange(event+1);
    elseif lengthchange(event+1)==0
        endchange=lengthchange(event);
    end
end

```

```

        else
            endchange=(lengthchange(event)-
lengthchange(event+1))/2;
        end

temponofftimes(10,event)=temponofftimes(10,event)+endchange;
    temponofftimes(9,(event+1))=temponofftimes(10,event);
    %Extract amplitudes for events longer than Tr at the same
time
    if fluorophores(fluor).onofftimes_fr(3,event)>Tr
        eventstart=temponofftimes(4,event)+1;
        eventend=temponofftimes(5,event);

fluorophores(fluor).onofftimes_fr(7,event)=mean(fluorophores(fluor)
).CKintcorr(eventstart:eventend));
        end
    end
    %Extract amplitude for final event if long enough
    if fluorophores(fluor).onofftimes_fr(3,nevents)>Tr
        eventstart=temponofftimes(4,nevents)+1;
        eventend=nframes;

fluorophores(fluor).onofftimes_fr(7,nevents)=mean(fluorophores(fluor)
or).CKintcorr(eventstart:eventend));
        end

        temponofftimes(11,:)=temponofftimes(10,:)-temponofftimes(9,:);
        fluorophores(fluor).onofftimes_fr(4,:)=temponofftimes(11,:);
        fluorophores(fluor).onofftimes_fr(5,:)=temponofftimes(9,:);
        fluorophores(fluor).onofftimes_fr(6,:)=temponofftimes(10,:);

        %Determine the false event detection rate (per frame)

offsects=fluorophores(fluor).CKintcorr(fluorophores(fluor).det_thr
esh==0);
        fluorophores(fluor).falserate=length(find(offsects<(-
(A0/2))))/length(offsects);
        if fluorophores(fluor).bkgdsub==1
            falserates(falsen)=fluorophores(fluor).falserate;
            falsen=falsen+1;
        end

    end

if isempty(Tr)
    [ avg_bkgd ] = extractfluordata( fluorophores, 'bkgdsub', 1 );
    avg_falserate=mean(avg_bkgd(6,:));
else
    avg_falserate=mean(falserates);
end

end

function [estimates, model] = fitline(xdata, ydata)
% Call fminsearch with a random starting point.
start_point = rand(1, 4);
model = @line;
options=optimset('Display','off');
[estimates] = fminsearch(model, start_point, options);

```

```

% line accepts curve parameters as inputs, and outputs sse,
% the sum of squares error for m*xdata+c-ydata,
% and the FittedCurve. FMINSEARCH only needs sse, but we want
% to plot the FittedCurve at the end.
function [sse, FittedCurve] = line(params)
    a = params(1);
    b = params(2);
    c = params(3);
    d = params(4);
    FittedCurve = a.*exp(-b*xdata+c)+d;
    ErrorVector = FittedCurve - ydata;
    sse = sum(ErrorVector .^ 2);
end
end

function FittedCurve = linefinal(xdata, params)
a = params(1);
b = params(2);
c = params(3);
d = params(4);
FittedCurve=a.*exp(-b*xdata+c)+d;
end

```

resolutionlimit

Inputs

fluorophores: array of structures containing details of fluorophore attributes (see Fluorophore structure fields)

tres: resolution limit to be imposed on the event duration record in frames

framerate: frame rate at which the image series was recorded

Tr: rise time of the Chung Kennedy filter used to smooth the raw intensity traces

A0: average signal above the background intensity level

Outputs

fluorophores: array of structures containing details of fluorophore attributes, including the newly calculated onofftimes_res field

```

function [ fluorophores ] = resolutionlimit( fluorophores, tres,
framerate, Tr, A0 )
%UNTITLED Summary of this function goes here
% Detailed explanation goes here
% Impose a consistent time resolution limit across all the on-off
time data
% Convert times in frames to a chosen time base.

% tres - time resolution limit in frames
% framerate - frame rate in chosen time base for final on off time
data

nfluors=size(fluorophores,2);
nframes=size(fluorophores(1).int,2);

for fluor=1:nfluors

    %Check there is onoff time data for the current fluorophore;
    if not,

```

```

    %pass on to the next
    if fluorophores(fluor).bkgdsub==0
        continue
    end

    nevents1=size(fluorophores(fluor).onofftimes_fr,2);
    temponoff=zeros(12,nevents1);

    %Populate the temporary onoff data array
    temponoff(1:2,:)=fluorophores(fluor).onofftimes_fr(1:2,:);
    temponoff(3:4,:)=fluorophores(fluor).onofftimes_fr(5:6,:);
    temponoff(5,:)=fluorophores(fluor).onofftimes_fr(4,:);
    temponoff(6,:)=fluorophores(fluor).onofftimes_fr(7,:);
    temponoff(7,1)=1;
    temponoff(8,1)=temponoff(2,1);
    lastlongevent=find(temponoff(5,:)>tres,1,'last');

    %Find events shorter than tres and concatenate with longer
    events
    nextevent=1;
    %Special case for short first event
    curreventlength=temponoff(5,1);
    if curreventlength>tres
        nextevent=nextevent+1;
        temponoff(7,nextevent)=temponoff(7,1)+1;
        temponoff(9,1)=temponoff(6,1);
        temponoff(10:12,1)=temponoff(3:5,1);
    else
        nextno=2;
        while temponoff(5,nextno)<=tres
            nextno=nextno+1;
        end
        nexttype=temponoff(2,nextno);
        nextamp=temponoff(6,nextno);
        shortlength=sum(temponoff(5,1:(nextno-1)));
        if shortlength<=tres
            temponoff(7,1:nextno)=1;
            temponoff(8,1:nextno)=temponoff(2,nextno);
            temponoff(9,1:nextno)=nextamp;
            temponoff(10,1:nextno)=0;
            temponoff(11,1:nextno)=temponoff(4,nextno);
            temponoff(12,1:nextno)=temponoff(11,nextno)-
            temponoff(10,nextno);
            nextevent=nextno+1;
        else
            shortamp=mean(fluorophores(fluor).intcorr(1:floor(shortlength)));
            if shortamp>(A0/2)
                %Short event cluster classed as 'on'
                temponoff(7,2:(nextno-1))=1;
                temponoff(7,nextno)=2;
                temponoff(8,1:(nextno-1))=1;
                temponoff(8,nextno)=temponoff(2,nextno);
                if floor(shortlength)>Tr
                    temponoff(9,1:(nextno-1))=shortamp;
                end
                temponoff(9,nextno)=temponoff(6,nextno);
                temponoff(10,nextno)=temponoff(3,nextno);
                temponoff(11,1:(nextno-1))=temponoff(3,nextno);
            end
        end
    end

```

```

        temponoff(11,nextno)=temponoff(4,nextno);
        temponoff(12,1:(nextno-1))=shortlength;
        temponoff(12,nextno)=temponoff(5,nextno);
        nextevent=nextno+1;
    else
        %Short event cluster classed as 'off'
        if nexttype==0
            temponoff(7,2:nextno)=1;
            temponoff(8,1:nextno)=0;
            temponoff(9,1:nextno)=nextamp;
            temponoff(11,1:nextno)=temponoff(4,nextno);
            temponoff(12,1:nextno)=temponoff(11,nextno)-
temponoff(10,nextno);
            nextevent=nextno+1;
        else
            temponoff(7,2:(nextno-1))=1;
            temponoff(7,nextno)=2;
            temponoff(8,1:(nextno-1))=0;
            temponoff(8,nextno)=temponoff(2,nextno);
            if floor(shortlength)>Tr
                temponoff(9,1:(nextno-1))=shortamp;
            end
            temponoff(9,nextno)=temponoff(6,nextno);
            temponoff(10,nextno)=temponoff(3,nextno);
            temponoff(11,1:(nextno-
1))=temponoff(3,nextno);
            temponoff(11,nextno)=temponoff(4,nextno);
            temponoff(12,1:(nextno-1))=shortlength;
            temponoff(12,nextno)=temponoff(5,nextno);
            nextevent=nextno+1;
        end
    end
end
end

choicelog=zeros(2,nevents1);
for event=2:lastlongevent
    if ~isequal(event,nextevent)
        continue
    end
    curreventlength=temponoff(5,event);
    if curreventlength>tres
        nextevent=nextevent+1;
        temponoff(7,event)=temponoff(7,(event-1))+1;
        temponoff(8,event)=temponoff(2,event);
        temponoff(9,event)=temponoff(6,event);
        temponoff(10:12,event)=temponoff(3:5,event);
        choice=1;
    else
        prevno=event-1;
        prevtype=temponoff(8,prevno);
        nextno=event+1;
        while temponoff(5,nextno)<=tres
            nextno=nextno+1;
        end
        nexttype=temponoff(2,nextno);
        prevamp=temponoff(9,prevno);
        nextamp=temponoff(6,nextno);
        shortlength=sum(temponoff(5,event:(nextno-1)));
        if shortlength<=tres

```

```

        if prevtype==0
            if nexttype==0

temponoff(7,event:nextno)=temponoff(7,prevno);
                temponoff(8,event:nextno)=0;
                if prevamp~=0
                    if nextamp~=0
                        newamp=(prevamp+nextamp)/2;
                    else
                        newamp=prevamp;
                    end
                elseif nextamp~=0
                    newamp=nextamp;
                else
                    newamp=0;
                end
                temponoff(9,prevno:nextno)=newamp;

newlength=temponoff(12,prevno)+sum(temponoff(5,event:nextno));
                temponoff(12,prevno:nextno)=newlength;

temponoff(10,event:nextno)=temponoff(10,prevno);

temponoff(11,prevno:nextno)=temponoff(10,prevno)+newlength;
                nextevent=nextno+1;
                choice=2;
            else
                temponoff(7,event:(nextno-
1))=temponoff(7,prevno);
                    temponoff(7,nextno)=temponoff(7,prevno)+1;
                    temponoff(8,event:(nextno-1))=0;
                    temponoff(8,nextno)=1;
                    temponoff(9,event:(nextno-1))=prevamp;
                    temponoff(9,nextno)=nextamp;
                    temponoff(10,event:(nextno-
1))=temponoff(10,prevno);
                        temponoff(10,nextno)=temponoff(3,nextno);

newlength=temponoff(12,prevno)+shortlength;
                    temponoff(12,prevno:(nextno-1))=newlength;
                    temponoff(12,nextno)=temponoff(5,nextno);
                    temponoff(11,prevno:(nextno-
1))=temponoff(10,prevno)+newlength;
                        temponoff(11,nextno)=temponoff(4,nextno);
                        nextevent=nextno+1;
                        choice=3;
                end
            else
                if nexttype==1
                    temponoff(8,event:nextno)=1;
                    if prevamp~=0
                        if nextamp~=0
                            if abs(prevamp-nextamp)>(0.5*A0)
                                temponoff(7,event:(nextno-
1))=temponoff(7,prevno);

temponoff(7,nextno)=temponoff(7,event)+1;
                                    temponoff(9,event:(nextno-
1))=prevamp;
                                        temponoff(9,nextno)=nextamp;

```

```

newlength=temponoff(12,prevno)+shortlength;
                                temponoff(12,prevno:(nextno-
1))=newlength;
                                temponoff(10,event:(nextno-
1))=temponoff(10,prevno);

temponoff(10:12,nextno)=temponoff(3:5,nextno);
                                temponoff(11,prevno:(nextno-
1))=temponoff(10,prevno)+newlength;
                                nextevent=nextno+1;
                                choice=4;
                                else
                                    newamp=(prevamp+nextamp)/2;

temponoff(7,event:nextno)=temponoff(7,prevno);

temponoff(9,prevno:nextno)=newamp;

newlength=temponoff(12,prevno)+sum(temponoff(5,event:nextno));
temponoff(12,prevno:nextno)=newlength;

temponoff(10,event:nextno)=temponoff(10,prevno);

temponoff(11,prevno:nextno)=temponoff(10,prevno)+newlength;
                                nextevent=nextno+1;
                                choice=5;
                                end
                                %Have previous event amplitude,
but not next
                                %event amplitude
                                else
                                    newamp=prevamp;

temponoff(7,event:nextno)=temponoff(7,prevno);
                                temponoff(9,prevno:nextno)=newamp;

newlength=temponoff(12,prevno)+sum(temponoff(5,event:nextno));
temponoff(12,prevno:nextno)=newlength;

temponoff(10,event:nextno)=temponoff(10,prevno);

temponoff(11,prevno:nextno)=temponoff(10,prevno)+newlength;
                                nextevent=nextno+1;
                                choice=6;
                                end
                                %Don't have the previous event
amplitude, but do
                                %have the next event amplitude
                                elseif nextamp~=0
                                    newamp=nextamp;

temponoff(7,event:nextno)=temponoff(7,prevno);
                                temponoff(9,prevno:nextno)=newamp;

newlength=temponoff(12,prevno)+sum(temponoff(5,event:nextno));
                                temponoff(12,prevno:nextno)=newlength;

```



```

temponoff(10,event:nextno)=temponoff(10,prevno);

temponoff(11,prevno:nextno)=temponoff(10,prevno)+newlength;
                             nextevent=nextno+1;
                             choice=7;
                             else
                                 %Have neither previous nor next event
amplitude

newlength=temponoff(12,prevno)+sum(temponoff(5,event:nextno));

temponoff(7,event:nextno)=temponoff(7,prevno);

temponoff(9,prevno:nextno)=temponoff(9,prevno);
                             temponoff(12,prevno:nextno)=newlength;

temponoff(10,event:nextno)=temponoff(10,prevno);

temponoff(11,prevno:nextno)=temponoff(10,prevno)+newlength;
                             nextevent=nextno+1;
                             choice=8;
                             end
                             else
                                 %Prevtype==1, nexttype==0
                                 temponoff(7,event:(nextno-
1))=temponoff(7,prevno);
                                 temponoff(7,nextno)=temponoff(7,prevno)+1;
                                 temponoff(8,event:(nextno-1))=1;
                                 temponoff(8,nextno)=0;
                                 temponoff(9,event:(nextno-1))=prevamp;
                                 temponoff(9,nextno)=nextamp;
                                 temponoff(10,event:(nextno-
1))=temponoff(10,prevno);
                                 temponoff(10,nextno)=temponoff(3,nextno);

newlength=temponoff(12,prevno)+shortlength;
                             temponoff(12,prevno:(nextno-1))=newlength;
                             temponoff(12,nextno)=temponoff(5,nextno);
                             temponoff(11,prevno:(nextno-
1))=temponoff(10,prevno)+newlength;
                             temponoff(11,nextno)=temponoff(4,nextno);
                             nextevent=nextno+1;
                             choice=9;
                             end
                             end
                             else
                                 %Several short events occur together, longer than
tres
                                 eventstart=ceil(temponoff(3,event));
                                 eventend=floor(temponoff(4,(nextno-1)));

shortamp=mean(fluorophores(fluor).intcorr(eventstart:eventend));
if shortamp>(A0/2)
    currtype=1;
else
    currtype=0;
end

```

```

        types=[prevtype,currtype,nexttype];
        if isequal(types,[1,0,1]) ||
isequal(types,[1,1,1])...
            || isequal(types,[1,1,0]) ||
isequal(types,[0,1,1])...
            || isequal(types,[0,1,0])
            temponoff(7,event:(nextno-
1))=temponoff(7,prevno)+1;
            temponoff(7,nextno)=temponoff(7,event)+1;
            temponoff(8,event:(nextno-1))=currtype;
            temponoff(8,nextno)=nexttype;
            temponoff(9,nextno)=temponoff(6,nextno);
            temponoff(10,event:(nextno-
1))=temponoff(3,event);
            temponoff(10,nextno)=temponoff(3,nextno);
            temponoff(11,event:(nextno-
1))=temponoff(10,event)+shortlength;
            temponoff(11,nextno)=temponoff(4,nextno);
            temponoff(12,event:(nextno-1))=shortlength;
            temponoff(12,nextno)=temponoff(5,nextno);
            nextevent=nextno+1;
            choice=10;
        elseif isequal(types,[1,0,0])

temponoff(7,event:nextno)=temponoff(7,prevno)+1;
            temponoff(8,event:nextno)=currtype;
            temponoff(9,event:nextno)=temponoff(6,nextno);
            temponoff(10,event:nextno)=temponoff(3,event);

temponoff(11,event:nextno)=temponoff(4,nextno);

temponoff(12,event:nextno)=shortlength+temponoff(5,nextno);
            nextevent=nextno+1;
            choice=11;
        elseif isequal(types,[0,0,1])
            temponoff(7,event:(nextno-
1))=temponoff(7,prevno);
            temponoff(7,nextno)=temponoff(7,(nextno-1))+1;
            temponoff(8,event:(nextno-1))=currtype;
            temponoff(8,nextno)=nexttype;
            temponoff(9,event:(nextno-
1))=temponoff(9,prevno);
            temponoff(9,nextno)=temponoff(6,nextno);
            temponoff(10,event:(nextno-
1))=temponoff(10,prevno);
            temponoff(10,nextno)=temponoff(3,nextno);
            temponoff(11,prevno:(nextno-
1))=temponoff(3,nextno);
            temponoff(11,nextno)=temponoff(4,nextno);
            temponoff(12,prevno:(nextno-
1))=temponoff(12,prevno)+shortlength;
            temponoff(12,nextno)=temponoff(5,nextno);
            nextevent=nextno+1;
            choice=12;
        elseif isequal(types,[0,0,0])
            temponoff(7,event:nextno)=temponoff(7,prevno);
            temponoff(8,event:nextno)=currtype;
            if prevamp~=0
                if nextamp~=0
                    newamp=(prevamp+nextamp)/2;

```

```

        else
            newamp=prevamp;
        end
        elseif nextamp~=0
            newamp=nextamp;
        else
            newamp=0;
        end
        temponoff(9,prevno:nextno)=newamp;

temponoff(10,event:nextno)=temponoff(10,prevno);

temponoff(11,prevno:nextno)=temponoff(4,nextno);

temponoff(12,prevno:nextno)=temponoff(12,prevno)+shortlength+tempo
noff(5,nextno);

        nextevent=nextno+1;
        choice=13;
    else
        disp(['New types=',num2str(types),' fluorno
',num2str(fluor),' eventno ',num2str(event)])
    end
end
end

choicelog(1,event)=event;
choicelog(2,event)=choice;

end

keep= choicelog(1,:)~=0;
choicelog=choicelog(:,keep);
%Special case for last pulse
if nextevent<=nevents1
    event=nextevent;
    shortlength=sum(temponoff(5,event:nevents1));
    prevno=event-1;
    prevtype=temponoff(8,prevno);
    prevamp=temponoff(9,prevno);
    if shortlength<=tres
        temponoff(7,event:nevents1)=temponoff(7,prevno);
        temponoff(8,event:nevents1)=temponoff(8,prevno);
        temponoff(9,event:nevents1)=prevamp;
        temponoff(10,event:nevents1)=temponoff(10,prevno);
        temponoff(11,prevno:nevents1)=nframes;

temponoff(12,prevno:nevents1)=temponoff(12,prevno)+shortlength;
    else
        eventstart=ceil(temponoff(3,event));
        eventend=nframes;

shortamp=mean(fluorophores(fluor).intcorr(eventstart:eventend));
        if shortamp>(A0/2)
            currtype=1;
        else
            currtype=0;
        end
        if prevtype==0 && currtype==0
            temponoff(7,event:nevents1)=temponoff(7,prevno);

```

```

        temponoff(9,event:nevents1)=prevamp;
        temponoff(10,event:nevents1)=temponoff(10,prevno);
        temponoff(11,prevno:nevents1)=nframes;

temponoff(12,prevno:nevents1)=temponoff(12,prevno)+shortlength;
    else
        temponoff(7,event:nevents1)=temponoff(7,prevno)+1;
        temponoff(8,event:nevents1)=currtype;
        temponoff(10,event:nevents1)=temponoff(3,event);
        temponoff(11,event:nevents1)=nframes;
        temponoff(12,event:nevents1)=shortlength;
    end
end
end

%Populate onofftimes_res
nevents2=temponoff(7,nevents1);
temponoff1=zeros(12,nevents2);
for event=1:nevents2
    save=find(temponoff(7,:)==event,1,'last');
    temponoff1(:,event)=temponoff(:,save);
end

fluorophores(fluor).onofftimes_res=zeros(9,nevents2);
fluorophores(fluor).onofftimes_res(1,:)=temponoff1(1,:);
fluorophores(fluor).onofftimes_res(2,:)=temponoff1(7,:);
fluorophores(fluor).onofftimes_res(3,:)=temponoff1(8,:);
fluorophores(fluor).onofftimes_res(4:5,:)=temponoff1(10:11,:);

fluorophores(fluor).onofftimes_res(6:8,:)=temponoff1(10:12,:)/frame
rate;
    fluorophores(fluor).onofftimes_res(9,:)=temponoff1(9,:);

end

end

```

MLfit2

Inputs

fluorophores: array of structures containing details of fluorophore attributes (see Fluorophore structure fields)

falsestatethresh: threshold value used to ignore intensity traces for which the false rate exceeds falsestatethresh (units of number of false events per frame)

tres: resolution limit imposed on the event duration record in frames

framerate: frame rate at which the image series was recorded

graphics: indicates whether to display histograms of event durations with MLE fit for each fluorophore (set to 0 to suppress graphics or 1 to display graphics)

Outputs

fitstruct: array of structures containing details of exponential fits to on and off event durations for each fluorophore

Key to fields in fitstruct

fluorno: fluorophore identifier number as assigned by LM_emitters (see 7.3. Appendix A:)

ontimes: list of on event durations

offtimes: list of off event durations

MLEests1: m by n by p array holding maximum likelihood estimates for a single exponential distribution fit to on times (p = 1) and off times (p = 2)

MLEests2: m by n by p array holding maximum likelihood estimates for a double exponential distribution fit to on times (p = 1) and off times (p = 2)

maxL: m by n by p array holding the maximum log(likelihood) for a single exponential distribution fit (m = 1) and a double exponential distribution fit (m = 2), and the probability that fitting the event duration distribution with two rather than one exponentials would produce, by chance, an improvement in fit equal to or greater than that observed, for on times (p = 1) and for off times (p = 2)

expsses: m by n by p array holding normalized residuals and exit flags of single and double exponential fits to on times (p = 1) and off times (p = 2)

expests1: m by n by p array holding estimates from single exponential fits to on times (p = 1) and off times (p = 2)

expests2: m by n by p array holding estimates from double exponential fits to on times (p

Custom functions called by MLfit2

make_loghist

```
function [ fitstruct ] = MLfit2( fluorophores, falseratethresh,
tres, framerate, graphics )
%UNTITLED Summary of this function goes here
%   Uses empirical fits of mixtures of exponentials to generate
starting
%   parameter values for MLE

nfluors=size(fluorophores,2);

%Create struct to return all fit values
%All off time equivalent fit values are stored 'behind' the on
values (i.e.
%in the p=2 layer of an m by n by p array)
%ontimes and offtimes are the curtailed distributions used for
fitting
%(i.e. with long duration outliers removed)
%maxL in this struct has [maxL1;maxL2;llrprob]
%expsses has [sse1,flag1;sse2,flag2]
fitstruct=struct('fluorno',cell(1,nfluors),'ontimes',cell(1,nfluor
s),...
    'offtimes',cell(1,nfluors),'MLEests1',cell(1,nfluors),...
    'MLEests2',cell(1,nfluors),'maxL',cell(1,nfluors),'expsses',cell(1
,nfluors),...
    'expests1',cell(1,nfluors),'expests2',cell(1,nfluors));

tres_time=tres/framerate;
binwidth=1/framerate;
```

```

for fluor=1:nfluors

    fitstruct(fluor).fluorno=fluorophores(fluor).fluorno;
    %Check on off times have been extracted for the fluorophore,
    and
    %whether the false event rate is acceptable. Pass on to the
    next if not
    if fluorophores(fluor).bkgdsub==0 ||
    fluorophores(fluor).falserate>falseratethresh
        continue
    end
    disp(fluor)

    %Extract on and off times

    ontimes=fluorophores(fluor).onofftimes_res(8,fluorophores(fluor).o
    nofftimes_res(3,:)==1);

    offtimes=fluorophores(fluor).onofftimes_res(8,fluorophores(fluor).
    onofftimes_res(3,:)==0);
    %Discard first and last events (which are possibly incomplete)
    if fluorophores(fluor).onofftimes_res(3,1)==1
        ontimes(1)=[];
    else
        offtimes(1)=[];
    end
    if fluorophores(fluor).onofftimes_res(3,end)==1
        ontimes(end)=[];
    else
        offtimes(end)=[];
    end
    onn=length(ontimes);
    offn=length(offtimes);

    %Organize data into histograms
    maxval=max([max(ontimes),max(offtimes)]);
    onmin=min(ontimes);
    offmin=min(offtimes);
    edges=0:binwidth:maxval;
    edges=[edges,edges(end)+binwidth];

    onstart=find(edges<onmin,1,'last');
    onend=find(edges<max(ontimes),1,'last');
    onxout=edges(onstart:onend)+binwidth/2;
    onfreq=hist(ontimes,onxout);

    offstart=find(edges<offmin,1,'last');
    offend=find(edges<max(offtimes),1,'last');
    offxout=edges(offstart:offend)+binwidth/2;
    offfreq=hist(offtimes,offxout);

    %Check histograms aren't skewed by a few long-duration
    outliers
    onsingles=find(onfreq==1);
    onsinglesdiffs=diff(onsingles);
    outlier=find(onsinglediffs>5,1,'first');
    ontmax=0;
    if ~isempty(outlier)

```

```

        outlier=outlier+1;
        outlierbin=onsingles(outlier);
        oncutoff=onxout(onsingles(outlier-1))+((onxout(outlierbin)-
onxout(onsingles(outlier-1)))/2);
        discard= ontimes>oncutoff;
        ontimes(discard)=[];
        ontmax=oncutoff;

        %Recalculate histogram
        onn=length(ontimes);
        onend=find(edges<max(ontimes),1,'last');
        onxout=edges(onstart:onend)+binwidth/2;
        onfreq=hist(ontimes,onxout);
    end
    nonbins=length(onxout);

    offsingles=find(offfreq==1);
    offsinglediffs=diff(offsingles);
    outlier=find(offsinglediffs>5,1,'first');
    offtmax=0;
    if ~isempty(outlier)
        outlier=outlier+1;
        outlierbin=offsingles(outlier);
        offcutoff=offxout(offsingles(outlier-
1))+((offxout(outlierbin)-offxout(offsingles(outlier-1)))/2);
        discard= offtimes>offcutoff;
        offtimes(discard)=[];
        offtmax=offcutoff;

        %Recalculate histogram
        offn=length(offtimes);
        offend=find(edges<max(offtimes),1,'last');
        offxout=edges(offstart:offend)+binwidth/2;
        offfreq=hist(offtimes,offxout);
    end
    noffbins=length(offxout);

    fitstruct(fluor).ontimes=ontimes;
    fitstruct(fluor).offtimes=offtimes;

    %Perform Maximum Likelihood fitting
    %Likelihood intervals (tauhi and tauilo) are asymmetrical
(unlike
    %approximate standard deviations); they are the m-unit
likelihood
    %intervals where m=2, corresponding to an approximate
probability of
    %0.95. This is only the case for the single exponential with
one fit
    %parameter. For mixtures of exponentials, the confidence
intervals are
    %2stds (95% confidence interval).

    %Create a matrix to hold the maximum values for the
log(likelihood) for
    %each fit of the data
    %
        ontimes offtimes
    %maxL(3x2): [ onmaxL1 offmaxL1 ] fit with a single exponential
    %
        [ onmaxL2 offmaxL2 ] fit with mix of 2 exponentials

```

```

%           [ onmaxL3 offmaxL3 ] fit with mix of 3
exponentials
    maxL=zeros(3,2);
    sses=zeros(3,4); %Report values of normalized sum of square
errors for
    %empirical exponential fits and exitflags
    onoffexpcomps=zeros(4,4); %Report values of empirical
exponential fit estimates

    %Try a single exponential distribution
    %MLtemp1(3x1): [ tauhat ]
    %              [ tau0   ]
    %              [ tauhi  ]

    onMLtemp1=zeros(3,1);
    offMLtemp1=zeros(3,1);

    %Use MATLAB's MLE function with custom negative loglikelihood
function to calculate
    %the maximum-likelihood estimate of the mean lifetime and the
    %associated likelihood intervals
    optsmle=statset('mlecustorm');
    optsmle.Display='notify';
    optsmle.MaxIter=300;
    optsmle.MaxFunEvals=600;

    if ontmax==0
        [onMLtemp1(1,1),onMLtemp1(2:3,1)] =
mle(ontimes,'nloglf',@negloglik1,'start',mean(ontimes)-
tres_time,'options',optsmle);
        %Determine the maximum value for the log(likelihood) for a
single
        %exponential distribution
        maxL(1,1)=-negloglik1(onMLtemp1(1,1),ontimes,[],[]);
        %Calculate the total number of observations (including
those too
        %short to detect)
        onNtrue1=onn*exp(tres_time/onMLtemp1(1,1));
    else
        tmaxn=ontmax;
        [onMLtemp1(1,1),onMLtemp1(2:3,1)] =
mle(ontimes,'nloglf',@negloglik1rest,'start',mean(ontimes)-
tres_time,'options',optsmle);
        maxL(1,1)=-negloglik1rest(onMLtemp1(1,1),ontimes,[],[]);
        %Calculate the total number of observations (including
those too
        %short to detect and those which were lost in curtailing
the
        %histogram)
        onNtrue1=onn/(exp(-tres_time/onMLtemp1(1,1))-exp(-
ontmax/onMLtemp1(1,1)));
    end
    if offtmax==0
        [offMLtemp1(1,1),offMLtemp1(2:3,1)] =
mle(offtimes,'nloglf',@negloglik1,'start',mean(offtimes)-
tres_time,'options',optsmle);
        maxL(1,2)=-negloglik1(offMLtemp1(1,1),offtimes,[],[]);
        offNtrue1=offn*exp(tres_time/offMLtemp1(1,1));
    else

```



```

        tmaxn=offtmax;
        [offMLtemp1(1,1),offMLtemp1(2:3,1)] =
mle(offtimes,'nloglf',@negloglik1,'start',mean(offtimes)-
tres_time,'options',optsmle);
        maxL(1,2)=-negloglik1rest(offMLtemp1(1,1),offtimes,[],[]);
        offNtrue1=offn/(exp(-tres_time/offMLtemp1(1,1))-exp(-
offtmax/offMLtemp1(1,1)));
    end

    fitstruct(fluor).MLEests1=zeros(3,1,2);
    fitstruct(fluor).MLEests1(:,1,1)=onMLtemp1;
    fitstruct(fluor).MLEests1(:,1,2)=offMLtemp1;

    %Determine the parameter for a single exponential by least
squares
    %minimization
    start_point1=zeros(1,2);
    start_point1(1)=onn*(onxout(2)-onxout(1));
    start_point1(2)=mean(ontimes)-tres_time;
    if ontmax==0
        [onestimates1, onmodell, onexitflag1] = fitexp(onxout,
onfreq, start_point1);
        [onssel, onFittedCurve1 ] = onmodell(onestimates1);
    else
        [onestimates1, onmodell, onexitflag1] = fitexprest(onxout,
onfreq, start_point1, ontmax);
        [onssel, onFittedCurve1 ] = onmodell(onestimates1);
    end

    start_point1(1)=offn*(offxout(2)-offxout(1));
    start_point1(2)=mean(offtimes)-tres_time;
    if offtmax==0
        [offestimates1, offmodell, offexitflag1] = fitexp(offxout,
offfreq, start_point1);
        [offssel, offFittedCurve1 ] = offmodell(offestimates1);
    else
        [offestimates1, offmodell, offexitflag1] =
fitexprest(offxout, offfreq, start_point1, offtmax);
        [offssel, offFittedCurve1 ] = offmodell(offestimates1);
    end

    %Normalize the sse
    onssel=onssel/nonbins;
    offssel=offssel/noffbins;
    sses(1,:)=[onssel,onexitflag1,offssel,offexitflag1];
    onoffexpcomps(1,1:2)=onestimates1;
    onoffexpcomps(3,1:2)=offestimates1;

    %Try a distribution with two exponential components
    %MLtemp2(3x4): [ ahat1 ahat2 tauhat1 tauhat2 ]
    %               [ alo1 alo2 tauo1 tauo2 ]
    %               [ ahil ahil2 tauhil tauhil2 ]

    onMLtemp2=zeros(3,4);
    offMLtemp2=zeros(3,4);

    %Fit a double exponential by least squares minimization
    start_point2=zeros(1,4);
    start_point2(1)=onn*(onxout(2)-onxout(1));

```

```

start_point2(2)=0.5;
start_point2(3:4)=mean(ontimes)-tres_time;
options=optimset('lsqcurvefit');
options.Display='off';
options.MaxFunEvals=800;
options.TolFun=1e-4;
options.TolX=1e-4;
if ontmax==0
    [onestimates2,onsse2,~,onexitflag2] =
lsqcurvefit(@expfun2,
start_point2,onxout,onfreq,[0,0,0,0],[inf,1,inf,inf],options);
    onFittedCurve2=expfun2(onestimates2,onxout);
else
    tmaxn=ontmax;
    [onestimates2,onsse2,~,onexitflag2] =
lsqcurvefit(@expfun2rest,
start_point2,onxout,onfreq,[0,0,0,0],[inf,1,inf,inf],options);
    onFittedCurve2=expfun2rest(onestimates2,onxout);
end

start_point2(1)=offn*(offxout(2)-offxout(1));
start_point2(2)=0.5;
start_point2(3:4)=mean(offtimes)-tres_time;
if offtmax==0
    [offestimates2,offsse2,~,offexitflag2] =
lsqcurvefit(@expfun2,
start_point2,offxout,offfreq,[0,0,0,0],[inf,1,inf,inf],options);
    offFittedCurve2=expfun2(offestimates2,offxout);
else
    tmaxn=offtmax;
    [offestimates2,offsse2,~,offexitflag2] =
lsqcurvefit(@expfun2rest,
start_point2,offxout,offfreq,[0,0,0,0],[inf,1,inf,inf],options);
    offFittedCurve2=expfun2rest(offestimates2,offxout);
end

%Normalize the sse
onsse2=onsse2/nonbins;
offsse2=offsse2/noffbins;
sses(2,:)=[onsse2,onexitflag2,offsse2,offexitflag2];
onoffexpcomps(2,:)=onestimates2;
onoffexpcomps(4,:)=offestimates2;

%Calculate the maximum-likelihood estimates of the parameters
and their
%associated likelihood intervals using the estimates from the
empirical
%exponential fit as starting points
if onexitflag2>0
    if ontmax==0
        [onMLtemp2(1,[1,3,4]),onMLtemp2(2:3,[1,3,4])] =
mle(ontimes,'nloglf',@negloglik2,'start',onestimates2(2:4),'lowerb
ound',[0,0,0],'upperbound',[1,inf,inf],'options',optsmle);
        onMLtemp2(1,2)=1-onMLtemp2(1,1);
        twostd=onMLtemp2(1,1)-onMLtemp2(2,1);
        onMLtemp2(2:3,2)=[onMLtemp2(1,2)-
twostd;onMLtemp2(1,2)+twostd];
        %Determine the maximum value for the log(likelihood)
for a mixed
        %distribution with two exponential components

```

```

        maxL(2,1)=-
negloglik2(onMLtemp2(1,[1,3,4]),ontimes,[],[]);
        %Determine the total number of observations (including
those too
        %short to detect)
        onNtrue2=onn/(onMLtemp2(1,1)*exp(-
tres_time/onMLtemp2(1,3))+onMLtemp2(1,2)*exp(-
tres_time/onMLtemp2(1,4)));
        else
            tmaxn=ontmax;
            [onMLtemp2(1,[1,3,4]),onMLtemp2(2:3,[1,3,4])] =
mle(ontimes,'nloglf',@negloglik2rest,'start',onestimates2(2:4),'lo
werbound',[0,0,0],'upperbound',[1,inf,inf],'options',optsmle);
            onMLtemp2(1,2)=1-onMLtemp2(1,1);
            twostd=onMLtemp2(1,1)-onMLtemp2(2,1);
            onMLtemp2(2:3,2)=[onMLtemp2(1,2)-
twostd;onMLtemp2(1,2)+twostd];
            %Determine the maximum value for the log(likelihood)
for a mixed
            %distribution with two exponential components
            maxL(2,1)=-
negloglik2rest(onMLtemp2(1,[1,3,4]),ontimes,[],[]);
            %Determine the total number of observations (including
those too
            %short to detect and those which were lost in
curtailing the
            %histogram)
            onNtrue2=onn/(onMLtemp2(1,1)*(exp(-
tres_time/onMLtemp2(1,3))-exp(-ontmax/onMLtemp2(1,3)))+...
            +onMLtemp2(1,2)*(exp(-tres_time/onMLtemp2(1,4))-
exp(-ontmax/onMLtemp2(1,4))));
            end
        end

        if offexitflag2>0
            if offtmax==0
                [offMLtemp2(1,[1,3,4]),offMLtemp2(2:3,[1,3,4])] =
mle(offtimes,'nloglf',@negloglik2,'start',offestimates2(2:4),'lowe
rbound',[0,0,0],'upperbound',[1,inf,inf],'options',optsmle);
                offMLtemp2(1,2)=1-offMLtemp2(1,1);
                twostd=offMLtemp2(1,1)-offMLtemp2(2,1);
                offMLtemp2(2:3,2)=[offMLtemp2(1,2)-
twostd;offMLtemp2(1,2)+twostd];
                maxL(2,2)=-
negloglik2(offMLtemp2(1,[1,3,4]),offtimes,[],[]);
                offNtrue2=offn/(offMLtemp2(1,1)*exp(-
tres_time/offMLtemp2(1,3))+offMLtemp2(1,2)*exp(-
tres_time/offMLtemp2(1,4)));
                else
                    tmaxn=offtmax;
                    [offMLtemp2(1,[1,3,4]),offMLtemp2(2:3,[1,3,4])] =
mle(offtimes,'nloglf',@negloglik2rest,'start',offestimates2(2:4),'
lowerbound',[0,0,0],'upperbound',[1,inf,inf],'options',optsmle);
                    offMLtemp2(1,2)=1-offMLtemp2(1,1);
                    twostd=offMLtemp2(1,1)-offMLtemp2(2,1);
                    offMLtemp2(2:3,2)=[offMLtemp2(1,2)-
twostd;offMLtemp2(1,2)+twostd];
                    maxL(2,2)=-
negloglik2rest(offMLtemp2(1,[1,3,4]),offtimes,[],[]);
                    offNtrue2=offn/(offMLtemp2(1,1)*(exp(-

```

```

tres_time/offMLtemp2(1,3))-exp(-offtmax/offMLtemp2(1,3)))...
      +offMLtemp2(1,2)*(exp(-tres_time/offMLtemp2(1,4))-
exp(-offtmax/offMLtemp2(1,4))));
    end
  end

  fitstruct(fluor).MLEests2=zeros(3,4,2);
  fitstruct(fluor).MLEests2(:,:,1)=onMLtemp2;
  fitstruct(fluor).MLEests2(:,:,2)=offMLtemp2;
  fitstruct(fluor).expsses=zeros(2,2,2);
  fitstruct(fluor).expsses(:,:,1)=sses(1:2,1:2);
  fitstruct(fluor).expsses(:,:,2)=sses(1:2,3:4);
  fitstruct(fluor).expests1=zeros(1,2,2);
  fitstruct(fluor).expests1(:,:,1)=onoffexpcomps(1,1:2);
  fitstruct(fluor).expests1(:,:,2)=onoffexpcomps(3,1:2);
  fitstruct(fluor).expests2=zeros(1,4,2);
  fitstruct(fluor).expests2(:,:,1)=onoffexpcomps(2,:);
  fitstruct(fluor).expests2(:,:,2)=onoffexpcomps(4,:);

  %Compare log likelihood ratios to determine which fit is best
  llrprobs=zeros(2,1);
  onllr=maxL(2,1)-maxL(1,1);
  llrprobs(1)=chi2pdf(2*onllr,4);
  offllr=maxL(2,2)-maxL(1,2);
  llrprobs(2)=chi2pdf(2*offllr,4);

  fitstruct(fluor).maxL=zeros(3,1,2);
  fitstruct(fluor).maxL(1:2,1,1)=maxL(1:2,1);
  fitstruct(fluor).maxL(3,1,1)=llrprobs(1,1);
  fitstruct(fluor).maxL(1:2,1,2)=maxL(1:2,2);
  fitstruct(fluor).maxL(3,1,2)=llrprobs(2,1);

  if graphics==1
    %Plot normal histogram for ontimes (excluding very long
times which
    %were omitted from fitting)
    figure
    bar(onxout,onfreq)
    xlabel('On time (s)')
    ylabel(['Frequency per ',num2str(onxout(2)-onxout(1)),'
s'])
    hold on
    if ontmax==0
      onMLEcurve1=@(t) onNtrue1*(onxout(2)-onxout(1))*(exp(-
(t-tres_time)/onMLtemp1(1,1))./onMLtemp1(1,1));
      onMLEcurve2=@(t) onNtrue2*(onxout(2)-
onxout(1))*(onMLtemp2(1,1)*...
      exp(-
t/onMLtemp2(1,3))/onMLtemp2(1,3)+onMLtemp2(1,2)*exp(-
t/onMLtemp2(1,4))/onMLtemp2(1,4))...
      /(onMLtemp2(1,1)*exp(-
tres_time/onMLtemp2(1,3))+onMLtemp2(1,2)*exp(-
tres_time/onMLtemp2(1,4)));
    else
      onMLEcurve1=@(t) onNtrue1*(onxout(2)-onxout(1))*(exp(-
t/onMLtemp1(1,1))./...
      (onMLtemp1(1,1)*(exp(-tres_time/onMLtemp1(1,1))-
exp(-ontmax/onMLtemp1(1,1)))));
      onMLEcurve2=@(t) onNtrue1*(onxout(2)-

```

```

onxout(1)) * (onMLtemp2(1,1) .* exp(-
t/onMLtemp2(1,3))/onMLtemp2(1,3)+...
onMLtemp2(1,2) .* exp(-
t/onMLtemp2(1,4))/onMLtemp2(1,4)) ...
/ (onMLtemp2(1,1) .* (exp(-tres_time/onMLtemp2(1,3)) -
exp(-ontmax/onMLtemp2(1,3)))+...
onMLtemp2(1,2) .* (exp(-tres_time/onMLtemp2(1,4)) -
exp(-ontmax/onMLtemp2(1,4))));
end
yonMLEcurve1=onMLEcurve1(onxout);
yonMLEcurve2=onMLEcurve2(onxout);

plot(onxout,onFittedCurve1,onxout,onFittedCurve2,onxout,yonMLEcurve1,
onxout,yonMLEcurve2)
legend('Observations','Single exponential fit','Double
exponential fit','Single exponential MLE','Double exponential
MLE')

%Plot log scale histogram for ontimes (excluding very long
times)
figure
[ onlogfreq, onedges ] = make_loghist( ontimes, 10 );
xlabel('On time (s), log scale')
ylabel('Frequency, square root scale')
hold on
%Transform fitted curves to log scale
onNexp1=onestimates1(1)/(onxout(2)-onxout(1));
onNexp2=onestimates2(1)/(onxout(2)-onxout(1));
if ontmax==0
onexpcurve1=@(t,dlog) onNexp1*log(10).*dlog.*t .*
(exp(-(t-tres_time)/onestimates1(2))./onestimates1(2));
onexpcurve2=@(t,dlog)
onNexp2*log(10).*dlog.*t.*( (onestimates2(2)/onestimates2(3)) .*...
exp(-t/onestimates2(3)) + ((1-
onestimates2(2))/onestimates2(4)) .* exp(-t/onestimates2(4)))/...
(onestimates2(2) .* exp(-
tres_time/onestimates2(3)) + (1-onestimates2(2)) .* exp(-
tres_time/onestimates2(4))));
onMLEcurve1=@(t,dlog)
onNtrue1*log(10).*dlog.*t.*(exp(-(t-
tres_time)/onMLtemp1(1,1))./onMLtemp1(1,1));
onMLEcurve2=@(t,dlog)
onNtrue2*log(10).*dlog.*t.*(onMLtemp2(1,1)*...
exp(-
t/onMLtemp2(1,3))/onMLtemp2(1,3)+onMLtemp2(1,2)*exp(-
t/onMLtemp2(1,4))/onMLtemp2(1,4)) ...
/ (onMLtemp2(1,1)*exp(-
tres_time/onMLtemp2(1,3))+onMLtemp2(1,2)*exp(-
tres_time/onMLtemp2(1,4))));
else
onexpcurve1=@(t,dlog) onNexp1*log(10).*dlog.*t .*
(exp(-(t/onestimates1(2))./...
(onestimates1(2)*(exp(-tres_time/onestimates1(2))-
exp(-ontmax/onestimates1(2))))));
onexpcurve2=@(t,dlog)
onNexp2*log(10).*dlog.*t.*( (onestimates2(2)/onestimates2(3)) .*...
exp(-t/onestimates2(3)) + ((1-
onestimates2(2))/onestimates2(4)) .* exp(-t/onestimates2(4)))/...
(onestimates2(2) .* (exp(-
tres_time/onestimates2(3))-exp(-ontmax/onestimates2(3))) +...

```

```

        (1-onestimates2(2)) .* (exp(-
tres_time/onestimates2(4))-exp(-ontmax/onestimates2(4))));
        onMLEcurve1=@(t,dlog)
onNtrue1*log(10).*dlog.*t.*(exp(-t/onMLtemp1(1,1))./...
        (onMLtemp1(1,1)*(exp(-tres_time/onMLtemp1(1,1))-
exp(-ontmax/onMLtemp1(1,1))));
        onMLEcurve2=@(t,dlog)
onNtrue1*log(10).*dlog.*t.*(onMLtemp2(1,1).*exp(-
t/onMLtemp2(1,3))/onMLtemp2(1,3)+...
        onMLtemp2(1,2).*exp(-
t/onMLtemp2(1,4))/onMLtemp2(1,4)).)...
        /(onMLtemp2(1,1).*(exp(-tres_time/onMLtemp2(1,3))-
exp(-ontmax/onMLtemp2(1,3)))+...
        onMLtemp2(1,2).*(exp(-tres_time/onMLtemp2(1,4))-
exp(-ontmax/onMLtemp2(1,4))));
    end
    d=diff(onedges);
    dlog=10.^d;
    onmids=onedges(1:end-1)+(diff(onedges)/2);
    yonexpcurve1=onexpcurve1(onmids,dlog);
    yonexpcurve2=onexpcurve2(onmids,dlog);
    yonMLEcurve1=onMLEcurve1(onmids,dlog);
    yonMLEcurve2=onMLEcurve2(onmids,dlog);

    %Transform y-axis data for square root scale y-axis
    newfreq=@(y) sqrt(y)/2;
    newyaxis=@(y) (2*y).^2;

maxy=max([max(yonexpcurve1),max(yonexpcurve2),max(yonMLEcurve1),ma
x(yonMLEcurve2)]);
    maxoldy=ceil(newfreq(maxy));

    newonlogfreq=newfreq(onlogfreq);
    newyonexpl=newfreq(yonexpcurve1);
    newyonexp2=newfreq(yonexpcurve2);
    newyonMLE1=newfreq(yonMLEcurve1);
    newyonMLE2=newfreq(yonMLEcurve2);
    ytickspacing=ceil(maxoldy/10);
    oldyaxis=0:ytickspacing:maxoldy;

    bar(onedges,newonlogfreq,'histc')
    delete(findobj('marker','*'));

plot(onmids,newyonexpl,onmids,newyonexp2,onmids,newyonMLE1,onmids,
newyonMLE2)
    ylim([0 maxoldy])
    set(gca,'ytick',oldyaxis)
    set(gca,'yticklabel',num2cell(newyaxis(oldyaxis)))
    set(gca,'xscal','log')
    legend('Observations','Single exponential fit','Double
exponential fit','Single exponential MLE','Double exponential
MLE')

    %Plot normal histogram for offtimes (excluding very long
times which
    %were omitted from fitting)
    figure
    bar(offxout,offfreq)
    xlabel('Off time (s)')

```

```

        ylabel(['Frequency per ', num2str(offxout(2)-offxout(1)), '
s'])
        hold on
        if offtmax==0
            offMLEcurve1=@(t) offNtrue1*(offxout(2)-
offxout(1))*(exp(-(t-tres_time)/offMLtemp1(1,1))./offMLtemp1(1,1));
            offMLEcurve2=@(t) offNtrue2*(offxout(2)-
offxout(1))*(offMLtemp2(1,1)*...
                exp(-
t/offMLtemp2(1,3))/offMLtemp2(1,3)+offMLtemp2(1,2)*exp(-
t/offMLtemp2(1,4))/offMLtemp2(1,4))...
                /(offMLtemp2(1,1)*exp(-
tres_time/offMLtemp2(1,3))+offMLtemp2(1,2)*exp(-
tres_time/offMLtemp2(1,4)));
        else
            offMLEcurve1=@(t) offNtrue1*(offxout(2)-
offxout(1))*(exp(-t/offMLtemp1(1,1))./...
                (offMLtemp1(1,1)*(exp(-tres_time/offMLtemp1(1,1))-
exp(-offtmax/offMLtemp1(1,1)))));
            offMLEcurve2=@(t) offNtrue2*(offxout(2)-
offxout(1))*(offMLtemp2(1,1).*exp(-
t/offMLtemp2(1,3))/offMLtemp2(1,3)+...
                offMLtemp2(1,2).*exp(-
t/offMLtemp2(1,4))/offMLtemp2(1,4))...
                /(offMLtemp2(1,1).*(exp(-
tres_time/offMLtemp2(1,3))-exp(-offtmax/offMLtemp2(1,3)))+...
                offMLtemp2(1,2).*(exp(-tres_time/offMLtemp2(1,4))-
exp(-offtmax/offMLtemp2(1,4)))));
        end
        yoffMLEcurve1=offMLEcurve1(offxout);
        yoffMLEcurve2=offMLEcurve2(offxout);

plot(offxout, offFittedCurve1, offxout, offFittedCurve2, offxout, yoffM
LEcurve1, offxout, yoffMLEcurve2)
    legend('Observations', 'Single exponential fit', 'Double
exponential fit', 'Single exponential MLE', 'Double exponential
MLE')

    %Plot log scale histogram for offtimes (excluding very
long times)
    figure
    [ offlogfreq, offedges ] = make_loghist( offtimes, 10 );
    xlabel('Off time (s), log scale')
    ylabel('Frequency, square root scale')
    hold on
    %Transform fitted curves to log scale
    offNexp1=offestimates1(1)/(offxout(2)-offxout(1));
    offNexp2=offestimates2(1)/(offxout(2)-offxout(1));
    if offtmax==0
        offexpcurve1=@(t,dlog) offNexp1*log(10).*dlog.*t .*
(exp(-(t-tres_time)/offestimates1(2))./offestimates1(2));
        offexpcurve2=@(t,dlog)
offNexp2*log(10).*dlog.*t.*( (offestimates2(2)/offestimates2(3))
.*...
            exp(-t/offestimates2(3)) + ((1-
offestimates2(2))/offestimates2(4)) .* exp(-
t/offestimates2(4)))/...
            (offestimates2(2) .* exp(-
tres_time/offestimates2(3)) + (1-offestimates2(2)) .* exp(-
tres_time/offestimates2(4)));
    end

```

```

        offMLEcurve1=@(t,dlog)
offNtrue1*log(10).*dlog.*t.*(exp(-(t-
tres_time)/offMLtemp1(1,1))./offMLtemp1(1,1));
        offMLEcurve2=@(t,dlog)
offNtrue2*log(10).*dlog.*t.*(offMLtemp2(1,1)*...
exp(-
t/offMLtemp2(1,3))/offMLtemp2(1,3)+offMLtemp2(1,2)*exp(-
t/offMLtemp2(1,4))/offMLtemp2(1,4))...
/(offMLtemp2(1,1)*exp(-
tres_time/offMLtemp2(1,3))+offMLtemp2(1,2)*exp(-
tres_time/offMLtemp2(1,4)));
    else
        offexpcurve1=@(t,dlog) offNexp1*log(10).*dlog.*t .*
(exp(-(t/offestimates1(2))./...
(offestimates1(2)*(exp(-
tres_time/offestimates1(2))-exp(-offtmax/offestimates1(2)))));
        offexpcurve2=@(t,dlog)
offNexp2*log(10).*dlog.*t.*( (offestimates2(2)/offestimates2(3))
.*...
exp(-(t/offestimates2(3)) + ((1-
offestimates2(2))/offestimates2(4)) .* exp(-
t/offestimates2(4)))/...
(offestimates2(2) .* (exp(-
tres_time/offestimates2(3))-exp(-offtmax/offestimates2(3))) +...
(1-offestimates2(2)) .* (exp(-
tres_time/offestimates2(4))-exp(-offtmax/offestimates2(4)))));
        offMLEcurve1=@(t,dlog)
offNtrue1*log(10).*dlog.*t.*(exp(-(t/offMLtemp1(1,1))./...
(offMLtemp1(1,1)*(exp(-(tres_time)/offMLtemp1(1,1))-
exp(-offtmax/offMLtemp1(1,1)))));
        offMLEcurve2=@(t,dlog)
offNtrue1*log(10).*dlog.*t.*(offMLtemp2(1,1).*exp(-
t/offMLtemp2(1,3))/offMLtemp2(1,3)+...
offMLtemp2(1,2).*exp(-
t/offMLtemp2(1,4))/offMLtemp2(1,4))...
/(offMLtemp2(1,1).*exp(-
tres_time/offMLtemp2(1,3))-exp(-offtmax/offMLtemp2(1,3)))+...
offMLtemp2(1,2).*exp(-(tres_time)/offMLtemp2(1,4))-
exp(-offtmax/offMLtemp2(1,4))));
    end
    d=diff(offedges);
    dlog=10.^d;
    offmids=offedges(1:end-1)+(diff(offedges)/2);
    yoffexpcurve1=offexpcurve1(offmids,dlog);
    yoffexpcurve2=offexpcurve2(offmids,dlog);
    yoffMLEcurve1=offMLEcurve1(offmids,dlog);
    yoffMLEcurve2=offMLEcurve2(offmids,dlog);

    %Transform y-axis data for square root scale y-axis

offmaxy=max([max(yoffexpcurve1),max(yoffexpcurve2),max(yoffMLEcurv
e1),max(yoffMLEcurve2)]);
    offmaxoldy=ceil(newfreq(offmaxy));

    newofflogfreq=newfreq(offlogfreq);
    newyoffexpl=newfreq(yoffexpcurve1);
    newyoffexp2=newfreq(yoffexpcurve2);
    newyoffMLE1=newfreq(yoffMLEcurve1);
    newyoffMLE2=newfreq(yoffMLEcurve2);
    ytickspacing=ceil(offmaxoldy/10);

```



```

        offoldyaxis=0:ytickspacing:offmaxoldy;

        bar(offedges,newofflogfreq,'histc')
        delete(findobj('marker','*'));

plot(offmids,newyoffexpl,offmids,newyoffexp2,offmids,newyoffMLE1,of
fmids,newyoffMLE2)
    ylim([0 offmaxoldy])
    set(gca,'ytick',offoldyaxis)
    set(gca,'yticklabel',num2cell(newyaxis(offoldyaxis)))
    set(gca,'xscal','log')
    legend('Observations','Single exponential fit','Double
exponential fit','Single exponential MLE','Double exponential MLE')
end
end

function [estimates, model, exitflag] = fitexp(xdata, ydata,
start_point)
% Call fminsearch with the starting point provided.
model = @expfun;
options=optimset('Display','off');
[estimates,~,exitflag] = fminsearch(model, start_point, options);
% expfun accepts curve parameters as inputs, and outputs sse,
% the sum of squares error for A*exp(-lambda*xdata)-ydata,
% and the FittedCurve.
    function [sse, FittedCurve ] = expfun(params)
        Nd=params(1);
        tau = params(2);
        FittedCurve = Nd .* (exp(-(xdata-tres_time)/tau)./tau);
        ErrorVector = FittedCurve - ydata;
        sse = sum(ErrorVector .^ 2);
    end
end

function [estimates, model, exitflag] = fitexprest(xdata, ydata,
start_point, tmax)
% Call fminsearch with the starting point provided.
model = @expfun;
options=optimset('Display','off');
[estimates,~,exitflag] = fminsearch(model, start_point, options);
% expfun accepts curve parameters as inputs, and outputs sse,
% the sum of squares error for A*exp(-lambda*xdata)-ydata,
% and the FittedCurve.
    function [sse, FittedCurve ] = expfun(params)
        Nd=params(1);
        tau = params(2);
        FittedCurve = Nd .* (exp(-xdata/tau)./(tau*(exp(-
tres_time/tau)-exp(-tmax/tau))));
        ErrorVector = FittedCurve - ydata;
        sse = sum(ErrorVector .^ 2);
    end
end

function fval = negloglik1(params,data,cens,freq)
n=length(data);
sumn=sum(data-tres_time);
fval=n*log(1/params)-sumn/params;
fval=-fval;
end

```

```

function fval = negloglik1rest(params,data,cens,freq)
    tau=params;
    ft = exp(-data/tau) ./ (tau*(exp(-tres_time/tau)-exp(-
tmaxn/tau)));
    fval=sum(log(ft));
    fval=-fval;
end

function FittedCurve = expfun2(params, xdata)
    Nd = params(1);
    a1 = params(2);
    tau1 = params(3);
    tau2 = params(4);
    FittedCurve = Nd*((a1/tau1) .* exp(-xdata/tau1) + ((1-
a1)/tau2) .* exp(-xdata/tau2))/...
    (a1 .* exp(-tres_time/tau1) + (1-a1) .* exp(-
tres_time/tau2));
end

function FittedCurve = expfun2rest(params, xdata)
    Nd = params(1);
    a1 = params(2);
    tau1 = params(3);
    tau2 = params(4);
    FittedCurve = Nd*((a1/tau1) .* exp(-xdata/tau1) + ((1-
a1)/tau2) .* exp(-xdata/tau2))/...
    (a1 .* (exp(-tres_time/tau1)-exp(-tmaxn/tau1)) + (1-
a1) .* (exp(-tres_time/tau2)-exp(-tmaxn/tau2)));
end

function fval = negloglik2(params,data,cens,freq)
    a1=params(1);
    tau1=params(2);
    tau2=params(3);
    ft=(a1*exp(-data/tau1)/tau1+(1-a1)*exp(-
data/tau2)/tau2)...
    /(a1*exp(-tres_time/tau1)+(1-a1)*exp(-tres_time/tau2));
    fval=sum(log(ft));
    fval=-fval;
end

function fval = negloglik2rest(params,data,cens,freq)
    a1=params(1);
    tau1=params(2);
    tau2=params(3);
    ft=(a1.*exp(-data/tau1)/tau1+(1-a1).*exp(-
data/tau2)/tau2)...
    /(a1.*(exp(-tres_time/tau1)-exp(-tmaxn/tau1))+(1-
a1).*(exp(-tres_time/tau2)-exp(-tmaxn/tau2)));
    fval=sum(log(ft));
    fval=-fval;
end

end

```

sqrtx

```
function [ ydata ] = sqrtx( xdata, params )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here

y0=params(1);
A=params(2);

ydata = y0 + A*sqrt(xdata);

end
```

make_loghist

```
function [ freq, edges ] = make_loghist( rawdata, nbins )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here

% nbins - number of bins per order of magnitude interval

% Use 'histc' option with bar function for plotting resulting
% histogram,
% then change to logarithmic x-axis with set(gca,'xscal','log')

lowest=min(rawdata);
highest=max(rawdata);

%Determine upper and lower limits for the log histogram scale
nlow=0;
if lowest<1
    while (lowest*(10^nlow))<1
        nlow=nlow+1;
    end
elseif lowest>=10
    while (lowest*(10^nlow))>=10
        nlow=nlow-1;
    end
end

nhigh=0;
if highest<1
    while (highest*(10^nhigh))<1
        nhigh=nhigh+1;
    end
elseif highest>=10
    while (highest*(10^nhigh))>=10
        nhigh=nhigh-1;
    end
end
nhigh=nhigh-1;

nbins=nbins*abs(nhigh-nlow);
edges=logspace(-nlow,-nhigh,nbins+1);
freq=histc(rawdata,edges);

end
```